Structure and Memory:
A Computational Model of Storage, Gradience, and Priming

A Dissertation Presented

by

Aniello De Santo

to

The Graduate School

in Partial Fulfillment of the

Requirements

for the Degree of

Doctor of Philosophy

in

Linguistics

Stony Brook University

May 2020

Stony Brook University
The Graduate School

Aniello De Santo

We, the dissertation committee of the above candidate for the
Doctor of Philosophy degree, hereby recommend
acceptance of this dissertation.

Thomas Graf – Dissertation Advisor
Associate Professor, Department of Linguistics

John Frederick Bailyn – Chairperson of Defense
Professor, Department of Linguistics

Mark Aronoff
Professor, Department of Linguistics

Jon Sprouse
Professor, Department of Linguistics, University of Connecticut

This dissertation is accepted by the Graduate School

Eric Wertheimer
Dean of the Graduate School

Abstract of the Dissertation

Structure and Memory:
A Computational Model of Storage, Gradience, and Priming

by

Aniello De Santo

Doctor of Philosophy

in

Linguistics

Stony Brook University

2020

Theoretical linguists have long argued that humans' knowledge of language is internalized in the form of rich grammatical representations. Formalizing the connection between grammatical operations and cognitive processes would then make it possible for experimental data to inform syntactic theories of language knowledge and use.

This dissertation follows a line of research addressing these issues from a computational perspective. It does so by providing a transparent, interpretable link between structural representations and off-line processing behavior — the empirical observation that some sentences are overall harder to process than others. In particular, I expand on past literature arguing that a top-down parser for Minimalist grammars (Stabler, 1996, MGs) can be used to relate parsing behavior and grammatical structure to memory usage — thus asking to which degree the representations hypothesized by linguists are relevant to processing (Kobele et al., 2013; Gerth, 2015; Graf et al., 2017).

First, I explore the performance of the MG model on a variety of word order and relative clause processing asymmetries in Italian, thus demonstrating the sensitivity of the linking theory to detailed grammatical information. Then, I propose the MG parser as a good, non-probabilistic formal model of how gradient acceptability can be derived from categorical grammars. In doing so, I show how psycholinguistic data can address fundamental questions about the nature of grammatical knowledge. Finally, I evaluate the model's predictions for a variety of psycholinguistic phenomena known as syntactic priming effects, and propose possible extensions to the computational framework that explore the contributions of grammatical features to memory load.

By investigating the MG model's performance across this diverse array of processing phenomena, this dissertation adds further support to the psychological plausibility of fine-grained grammatical knowledge contributing to processing cost. It thus highlights the MG parsing model as a valuable, empirically grounded, theoretically insightful reframing of the Derivational Theory of Complexity (Miller and Chomsky, 1963).

You cannot hide in minimalist furniture!

— Felix Dawkins, *Orphan Black: Variation Under Nature*

Contents

List of Figures

List of Tables

Acknowledgements

Seven years ago or so, I was finishing a degree in computer science. I was also taking one-day trips to Rome to watch Chomksy give a talk, and reading neurolinguistics papers through the night. By the time I graduated with my Master's, I had come up with the weird idea that I should try and get into a PhD program in Linguistics. While far from random, that decision still came with a significant load of doubts and uncertainty. And yet, it was the best decision of my life.

I will be forever grateful to my advisor, Thomas Graf, for taking such a bet on me. If I am a somewhat decent scholar, I owe it to his patience and care. It is impossible to summarize the personal and intellectual influence he had on me (and yet, here I try). The way I think about linguistics will obviously be forever shaped by my years with him. He also taught me something that I had missed in all my years studying engineering: the beauty of mathematics, and the clarity that formalization can bring — when used with care. I am also grateful to him for leaving me the space to jump from topic to topic, and develop my own flavor of computational linguistics. He trusted me through my most scattered moments — sci-fi related disagreements notwithstanding.

This dissertation was defended during a weird historical moment, but I was lucky enough to end up with an incredible committee. Mark Aronoff has been, through the years, an infinite resource of knowledge about the field, and a great example of what it means to be a truly open-minded researcher. John Bailyn has taught me most of what I know about syntax. He is also who I aspire to be as a teacher — his Syntax II still the best class I had the privilege to sit through in my (so many) years as a college student. Finally, Jon Sprouse's influence on this work can be guessed by a quick look at my bibliography. What those references won't tell you though, are the deep insights that his questions bring.

A dissertation is really just one step in a long journey. The starting point, for me, was the meeting

with Andrea Moro. He introduced me to generative linguistics quite by accident, and got stuck with very, very long emails about logic and UG. Thank you, Andrea, for not letting that sparkle die. Thank you for being my first Maestro.

Similarly, I will be forever grateful to the whole Linguistics Department at Stony Brook for welcoming me and being my intellectual home. I will probably never know who was on that year's admission committee, but not many departments would have taken the risk of accepting an engineer with just a bit of self-taught syntax, and a lot of weird ideas. Everyone in the department played a big role in making Stony Brook more than a work-place, but some people deserve a few special words. In particular, Lori Repetti has been a mentor in many ways, way beyond being an inspiration as an amazing academic and linguist. Jeff Heinz clearly had a deep impact on my own research but, more importantly, he has been an example of what hard work and passion can accomplish.

One of the things I have bragged about at conferences since year one, is the atmosphere of mutual support that makes the PhD program at Stony Brook so unique. This was, of course, only possible thanks to the amazing people I had the privilege to study with. I want to thank Rob Pasternak, James Monette, and Brigi Fodor for welcoming me those first days on LI, and offering warm, geeky friendship through the years; Sophie Moradi, for reminding me of the importance of beauty and art while doing science; Chikako Takahashi and Alex Yeung, fellow foodies, for never judging and always understanding; Nazila Shafiei, for listening when it was needed; Hossep Dolatian, for shrugging off my self-loathing (and for Steven!); Ayla Karakas, for always asking fundamental questions (and reminding me that there is no shame in staying a bit emo); Jon Rawski, for the long, long debates (rants?) about language and cognition (and cheese, and wine, and much more). Thank you also to So Young Lee, Lei Liu, Chong Zhang, Veronica Miatto, Hyunah Baek, Hongchen Wu, Dakotah Lambert, SeoYoung Kim, Andrija Petrovic, and Grace Wivell.

There are not enough words to thank Alëna Aksënova: co-advisee, cohort member, collaborator, housemate, and friend. We did not end up killing each other, after-all. Here's to adding things to this list in the years to come.

I owe a lot to a long list of people outside of Stony Brook Linguistics. Thank you to Sanket Deshmukh, for sharing food and anxieties with me for many years. And big thanks to the people

that left their sign (and mail) on 26 Lake Grove: Abhishek De, Felix Keppler, Kabilan Ram Kumar, and Henrick Goldwurm.

Chiara Bonomi, Matteo Greco, Mariarosaria Musco, Antonio Tedeschi, Mariarosaria Marinozzi, and Lorenzo Sacchi deserve a special mention for never giving up on me — even across oceans, time-zones, and ignored messages. Thank you for putting up with my weirdness. You see things. And you understand.

Many others helped me — personally and professionally — through these strange, life-changing years. In no particular order, thanks to: Adam Jardine, Jane Chandlee, Kevin McMullin, Andrei Antonenko, John Drury, Marina Ermolaeva, Cassandra Jacobs, Ildi Szabo, Sabine Laszakovits, Mai Ha Vu, Jim Rogers, Bob Frank, Liina Pylkkanen, Antonio Picariello, Vincenzo Moscato, Flora Amato, Giancarlo Sperlì, Emanuela Pasi, Peter Cucé, Francesca Maglio, Mariarosaria Pelella, Martina Simonetti, Michela Marabini. I am also going to indulge my millennial self, and acknowledge that I owe a lot to the people of #LinguisticsTwitter. I was never good at conference networking, but they taught me the beauty of having a community of peers.

Without doubts, I am forgetting many important people thanks to whom I am where (and who) I am today. However, there are space limits to this thing, and memory has always been my enemy. But, as Diane Nguyen says: "*There are people that help you become the person you end up being, and you can be grateful for them, even if they were never meant to be in your life forever. I'm glad I knew you too*". I truly am glad, and deeply grateful.

Lastly, I saved this final paragraph for those who deserve my deepest gratitude. My siblings Andrea, Adelia, Alessia, and Tommaso. My grandmother Tommasina. And my parents, Massimo and Gilda. My thoughts go to you more often than I admit. I am truly sorry for all the questions you keep getting about what I do on this side of the ocean, and how linguistics and computer science are even related. Thank you for your support and love. Questa tesi è per voi.

Chapter 1

Introduction

It is a well-established fact that humans do not find all sentences equally easy to understand. While this might seem a trivial observation, the question of what makes some sentences more difficult to comprehend than others has been for decades at the center of debates in linguistics. Consider, the examples in (1) and (2):

(1) This is the cat that caught the rat that stole the cheese.

(2) This is the cat that the rat that stole the cheese caught.

Famously, English speakers find sentences like (1) easier to comprehend than sentences like (2) — as measured, for instance, by differences in reading times or comprehension accuracy (Miller and Chomsky, 1963, a.o.). Additionally, if we add syntactic material to sentences of the first type (3), these remain relatively easy to understand. On the contrary, difficulties in comprehending sentences of the second type (4) increase significantly.

(3) This is the dog that chased the cat that caught the rat that stole the cheese.

(4) This is the dog that the cat that the rat that stole the cheese caught chased.

From a theoretical point of view, the sentences above are notably different in terms of structural configurations: the way they are organized in “constituents of various type" (their *tree structure*; Chomsky, 1965). In this sense, sentences of the first kind are known as instances of left-branching

constructions, while sentences of the second kind are examples of center-embedding (or nested) constructions.

(5) [This is the dog that chased [the cat that caught [the rat that stole the cheese]]].

(6) [This is the dog that [the cat that [the rat that stole the cheese] caught] chased].

Notably, the oldest explanations for this *processing asymmetry* associate the difficulty of center-embedding sentences to the number of incomplete and nested syntactic relationships that must be maintained through the string (see Resnik, 1992, and references therein). While alternative accounts exist, most share the premise that there are fundamental differences in the kind of structures that need to be built in order to comprehend these sentences (Levy, 2013, a.o.).

Clearly, these considerations open the question of how much the structural information posited by theoretical linguists as part of our knowledge of language matters in determining processing behavior — and, vice-versa, whether experimental data can be used to inform our theories of linguistic knowledge.

Notoriously, theoretical linguists in the generative tradition are known to maintain the distinction between linguistic *competence* (a speaker's knowledge of the language) and *performance* (linguistic behavior). From this perspective, syntacticians often abstract away from performance factors to build their grammatical descriptions.

> Linguistic theory is concerned primarily with an ideal speaker-listener [...] who knows its language perfectly and is unaffected by such grammatically irrelevant conditions as memory limitations, distractions, shifts in attention and interest, and errors (random or characteristic) in applying his knowledge of the language in actual performance.
>
> (Chomsky, 1965, pg. 3)

In this respect, modern generative syntax developed as a cognitive enterprise trying to characterize the knowledge of such *ideal speakers*. Thus, the focus has been on ways to show how "the actual behavior of real native speakers converges on the ideal behavior predicted by our grammatical theory, as interfering performance factors are reduced" (Bresnan, 1982; Hale, 2011).

> The basic idea is that you can evaluate theories of grammar-based processing as to whether their behavior corresponds to the behavior of an ideal native speaker in the limit as the amount of available processing resources goes to infinity. Of course, the behavior of an ideal native speaker, one who knows his language perfectly and is not affected by restrictions of memory or processing time, lapses of attention, and so forth, is difficult to observe. But as psycholinguistic methods and technologies improve, we can imagine doing experiments in which we somehow vary the cognitive resources of real speakers and hearers, by removing distractions, giving them scratch-pad memories, etc. We can then take the limiting, asymptotic behavior of real speakers as approximations to the behavior of the ideal. A grammar-based processing model which, when given more and more computational resources, more and more accurately simulates the behavior of the ideal has the "ideal-convergent" property.
>
> (Kaplan, 1995, pg. 344)

Importantly, this stance has sometimes been interpreted as implying that grammatical theory has nothing to say to research in sentence processing, or that theoretical linguists are fundamentally uninterested in performance data. However, this comes from a misunderstanding of the position expressed above. In fact, the effect that sentence structure has on performance has been studied since the (modern) inception of generative grammar (Miller and Chomsky, 1963; Chomsky, 1965).

The competence/performance distinction is thus less of a cognitive claim, and more of a conceptual stance: a deep understanding of what constitutes humans' knowledge of language is necessarily *a first step* towards the characterization of a more complete cognitive system.

> [the] investigation of performance will proceed only so far as understanding of underlying competence permits. [...]
>
> [...] In general, it seems that the study of performance models incorporating generative grammars may be a fruitful study; furthermore, it is difficult to imagine any other basis on which a theory of performance might develop.
>
> (Chomsky, 1965, pg. 10-15)

However, while abstracting from performance factors seems to be a reasonable starting point in building an understanding of human linguistic abilities[1], the development of increasingly refined grammatical theories begs the question of whether they can be integrated with modern psycholinguistic research.

[1] In fact, separating the study of complex systems into multiple levels of exploration is a fundamental methodology in several areas of cognitive science (Marr et al., 1991; Hale, 2011).

> A realistic grammar should be psychologically real in the broad sense: it should contribute to the explanation of linguistic behavior and to our larger understanding of the human faculty of language.
>
> (Bresnan, 1978, pg. 58)

In particular, while it seems undebatable that grammatical information matters *to some extent* in deriving comprehension difficulty, a more fundamental question is to what degree the fine-grained structural representations postulated by modern syntactitians play a role during sentence processing.

In order to answer such questions, the field needs ways to test the predictions made by a pure competence grammar against empirical data. On top of a fully formalized theory of grammatical representations though, this requires a theory of how such representations are built from input, and a transparent link between structural complexity and processing difficulty. The claim at the core of this dissertation is that this is achievable through the use of explicit computational models.

1.1 Goals of the Dissertation

Specifying how grammatical structure drives processing cost in computational terms makes it possible to connect long-standing ideas about cognitive load in human language processing with explicit syntactic analyses in rigorous ways.

Importantly, computational methods do not *by themselves* provide a theoretical foundation to cognitive investigations. In particular, in the study of opaque cognitive systems, there is sometimes the risk to confuse insights into the mechanisms we are trying to understand, with details about the models we are using to understand them.[2] However, computational models *can* force researches to formulate their theoretical assumptions into explicit hypotheses. In this work, I show how a computational approach *can* provide an interpretable bridge between syntactic assumptions and

[2] Kaplan, somewhat flippantly, warns us about the "*compelling temptations or seductions*" one risks to fall into when applying computational methodologies to cognitive questions without explicitly committing to an independently motivated theoretical stance.

> [...] But then of course people starting using other random woolly types of computations. It was a reasonable move at the time but it led down the slippery slope (see Figure 1).

processing behavior (Joshi, 1990; Rambow and Joshi, 1994; Hale, 2011). This connection is supported by past work in theoretical syntax and in psycholinguistics, and will be detailed in Chapter 2.

Recent computational models of human sentence processing assume that the grammar is an abstract description of the representations built by the cognitive system during language processing (Hale, 2011; Lewis and Vasishth, 2005; Gerth, 2015). This is also the stance I take in this work.

In particular, I build on recent studies showing that the behavior of a parser for Minimalist grammars (MGs; Stabler, 1996, 2013) can link structural complexity to memory usage. This is an appealing approach, as it seems to successfully model processing preferences across a variety of phenomena cross-linguistically (Kobele et al., 2013; Gerth, 2015; Graf et al., 2017, a.o.). In particular, this takes the form of a specific implementation of Stabler (2013)'s top-down parser for MGs, coupled with a vast set of complexity metrics measuring how the tree traversal algorithm recruits memory during the processing of different types of sentences.

This dissertation argues that this parsing model represents an insightful, empirically grounded reframing of past theories trying to bridge the study of competence and the study of performance (e.g., the Derivational Theory of Complexity; Miller and Chomsky, 1963; Fodor and Garrett, 1967; Berwick and Weinberg, 1983).

I approach this claim from a threefold perspective. From one side, I investigate the extend of the model's sensitivity to detailed grammatical assumptions. In doing this, I add support to the claim that subtle structural differences modulate sentence complexity, and thus that grammatical knowledge can indeed help explain processing behavior. Secondly, I show that this bridge between experimental evidence and grammatical theory allows researchers to use psycholinguistic data to

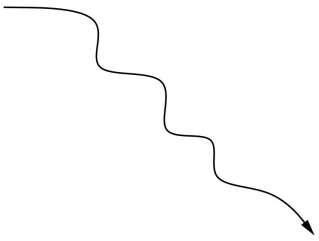


FIGURE 1 The slippery slope

(Kaplan, 1995, pg. 344-345)

address long-standing questions about the nature of human grammatical representations. Finally, I test the empirical limits of the model, asking new questions about the psychological plausibility of the linking theory and proposing possible extensions to the way memory usage is estimated.

In sum, by investigating the MG model's performance across this diverse array of processing phenomena, this dissertation adds further support to the psychological plausibility of the claim that fine-grained grammatical knowledge contributes to processing cost. Crucially, in approaching this problem I put aside questions about the time-course of processing effects (word-by-word, *online* processing), and I focus on characterizing how fine-grained syntactic details can modulate the overall complexity of a sentence (*off-line* processing). Moreover, I factor out the cost of ambiguity resolution, and assume that the parser deterministically pursues the correct structure-building steps. I will touch on these choices several times across the dissertation, but a first discussion of the rationale behind them is presented in Section 2.4 and Section 2.5.

1.2 Structure of the Dissertation

This dissertation is structured so that, after a general background chapter, the remaining three core chapters can be read somewhat independently. More precisely, each chapter is organized as follows.

Chapter 2: Conceptual and Technical Background The chapter can be divided in two. The first part (Sections 2.2 and 2.3) is a conceptual and historical overview of the relation between competence and performance in psycholinguistics. It also details the cognitive framework the parsing model draws on: memory-burden theories of processing complexity. The second part of the chapter (Section 2.4) is instead a technical overview of the computational approach used in the rest of the dissertation. It introduces Minimalist grammars, Stabler (2013)'s top-down parser, and a set of complexity metrics indexing memory usage.

Chapter 3: Italian Relative Clause Asymmetries This chapter explores the performance of the MG model on a variety of word order and relative clause processing asymmetries in Italian. The

Italian facts present some interesting challenges for the MG approach, as an attempt to account for processing complexity just in terms of structural factors. The results in this chapter thus further demonstrate the effectiveness of the model, and the sensitivity of the linking theory to detailed grammatical information. These results were initially presented in (De Santo, 2019).

Chapter 4: Gradience In this chapter, I argue that the MG parser provides a good formal framework to study how gradient acceptability can be derived from categorical grammars. In doing so, I show how psycholinguistic data can be used to address questions about the nature of grammatical knowledge. Thus, I propose the MG parser as one of the first quantitative models of how processing factors and fine-grained grammatical structure conspire to modulate sentence acceptability. A reduced version of these results appeared in (De Santo, 2020).

Chapter 5: Priming This chapter argues that, to truly understand how current theories of grammar affect processing behavior, it is crucial to investigate the role played by grammatical features in driving processing cost (Zhang, 2017). However, as encoded by the complexity metrics used so far in the literature, the link from MG tree structures to processing behavior is too coarse to capture features as an essential component of syntactic representations. Starting from these considerations, I evaluate the model's predictions for a variety of psycholinguistic phenomena known as *syntactic priming effects*, and propose extensions to the computational framework that explore the contributions of grammatical features to memory load.

Chapter 6: Conclusion This chapter concludes by reviewing successes and failures of the model. It also discusses suggestions for further research.

Chapter 2

Background

2.1 Introduction

This dissertation follows a long line of research trying to characterize how grammatical knowledge is applied in perceiving syntactic structure. In particular, while there is little doubt that underlying syntactic representations affect language use, it is unclear to what extent the richly detailed grammars postulated by modern theoreticians matter in studying processing behavior.

In this sense, computational models grounded in rich grammatical formalisms can provide a transparent, interpretable linking theory between syntactic assumptions and processing complexity. Thus, they can be used to explore whether — and to which degree — the structural representations hypothesized by theoretical linguists are relevant to sentence processing.

This chapter sets up the modeling approach used in this dissertation to address these questions, and can be conceptually divided into two parts. In the first half of the chapter, I review a long standing debate about the role of grammatical knowledge in the study of language processing (Section 2.2), and discuss the importance of theories of memory burden in psycholinguistics (Section 2.3). Section 2.4 is instead a technical introduction to the details of the computational model: a top-down parser for Minimalist grammars coupled with a set of complexity metrics measuring memory usage (Stabler, 2013; Kobele et al., 2013; Gerth, 2015; Graf et al., 2017).

Importantly, consistently with previous work on this model, I focus on the relation between structural representations and *off-line* processing behavior — the empirical observation that some

sentences are overall harder to process than others. Thus, questions about the time-course of processing complexity (*on-line* processing) are beyond the scope of this dissertation. The reader is referred to Section 2.5 for a brief discussion of these issues.

2.2 Grammatical Knowledge and Sentence Processing

Theoretical syntactitians have long argued that sentences hide complex hierarchical structures, and that a speaker's knowledge of the language is internalized in the form of rich grammatical representations. As mentioned in Chapter 1, a strong tenant of linguists in the generative tradition is then the importance of distinguishing linguistic *competence* (the knowledge of such grammatical representations) from linguistic *performance*:

> Linguistic theory is concerned primarily with an ideal speaker-listener [...] who knows its language perfectly, and is unaffected by such grammatically irrelevant conditions as memory limitations, distractions, shifts of attention, and interest, and errors (random or characteristic) in applying his knowledge of language in actual performance. [...] We thus make a fundamental distinction between competence (the speaker-hearer's knowledge of his language) and performance (the actual use of language in concrete situations). Only under the idealization set forth in the preceding paragraph is performance a direct reflection of competence.
>
> (Chomsky, 1965, pg. 3)

The fact that, in studying grammatical characterizations, syntactitians often abstract away from performance considerations has in the past led to the misconception that grammatical theory is irrelevant to how humans process linguistic input (sentence processing), or that psycholinguistic research into language use is irrelevant to the development of grammatical theories (cf. Kush and Dillon, To appear). On the contrary, the relation between language knowledge and language use has been at the core of linguistic inquiry since the early days of generative grammar.

> The fundamental fact that must be faced in any investigation of language and linguistics behavior is the following: a native speaker of a language has the ability to comprehend an immense number of sentences that he had never previously heard and to produce, on the appropriate occasion, novel utterances that are similarly

> understandable to other native speakers. The basic questions that must be asked are the following:
>
> 1. What is the precise nature of this ability?
> 2. How is it put to use?
> 3. How does it arise in the individual?
>
> (Miller and Chomsky, 1963, pg. 271)

In fact, generative linguists consider the grammars they build as *the* core component of models of language use (Berwick and Weinberg, 1982, 1983), and argue that formulating a precise theory of grammatical knowledge is a necessary step in addressing questions about human language processing mechanisms.

> This methodological slant should not, of course, be taken as implying that the investigation of grammar should have little contact with theories of language use or language acquisition, or, worse yet, that a complete understanding of language ends with the study of grammar. It simply claims that a proper way to *begin* the study of language is to start with a characterization of what that knowledge is — in short, with a theory of grammar.
>
> (Berwick and Weinberg, 1982, pg. 165-166)

These considerations, of course, lead to the question of how exactly should a theory of grammar be incorporated into a plausible model of language use — what Bresnan calls the *realization problem* (Halle et al., 1978, Chapter 1). In particular, if we view the grammar as describing the representations that should be computed by the sentence processing system (the parser), there are several ways in which the relation between grammatical theories and processing mechanisms can be specified.

In the past, the correct approach to characterizing such relation has been the object of extensive debates (the reader is referred to Berwick and Weinberg, 1982, 1983; Stabler, 1984; Berwick and Weinberg, 1985, for a classic example of such discussions). A common assumption in the early days of transformational grammar was that grammatical principles should guide processing strategies, with parsing mechanisms somehow mirroring the rules of the grammar. This is what Berwick and Weinberg (1983) refer to as the *Type Transparency Hypothesis* which, in its strongest

interpretation, demands a direct relation between "*the theoretical objects of grammar and those of parsing*". This is certainly an appealing view, as it would allow for experimental data from sentence processing experiments to immediately bear on theoretical hypotheses, opening syntactic theory to a whole new source of evidence.

> [...] the grammatical realization problem can clarify and delimit the grammatical characterization problem. We can narrow the class of possible theoretical solutions by subjecting them to experimental psychological investigation as well as to linguistic investigation.
>
> (Bresnan, 1978, pg. 59)

One of the most straightforward attempts to this line of investigation took the form of the *Derivational Theory of Complexity* (DTC; Miller and Chomsky, 1963; Miller and McKean, 1964), inspiring a significant amount of experimental work testing the ability of transformational analyses to predict the processing complexity of syntactic constructions (Levelt and Barnas, 1974; Fodor et al., 1974). However, evaluating whether such a direct connection between parsing operations and grammatical rules is cognitively plausible turned out to be more challenging than early enthusiasm lead researches to believe. As a consequence, most sentence processing research nowadays is less concerned with details of specific grammatical analyses, and more oriented towards characterizing processing mechanisms at a level that abstracts over very fine-grained grammatical details (Kush et al., 2018).

In what follows, I briefly review the ideas behind the DTC, and the validity of the arguments that lead to its demise. I argue that the core claims of Miller and Chomsky's proposal are in fact still relevant today. In particular, what seems to be needed is a model formulating specific hypotheses on the link between grammatical representations and parsing algorithms, to obtain predictions that are at the right level of resolution to be compared against experimental data. This is exactly the model that I will present in Section 2.4

2.2.1 The Derivational Theory of Complexity

Miller and Chomsky (1963)'s Derivational Theory of Complexity (DTC) is probably the simplest theory of how a direct relationship between grammar and parser could be realized.

> The psychological plausibility of a transformational model of the language user would be strengthened, of course, if it could be shown that our performance on tasks requiring an appreciation of the structure of transformed sentences is some function of the nature, number and complexity of the grammatical transformations involved
>
> (Miller and Chomsky, 1963, pg. 476).

In its strongest interpretation, the DTC proposed a *one-to-one mapping* between the processing complexity of a sentence and the length of its derivation by the grammar. In other words, the DTC associated a specific cognitive cost with the number of syntactic operations needed to derive a sentence: the processing complexity of a sentence could thus be accounted for by the number of *transformations* involved in that sentence's derivation (i.e., by the length of the derivation process).[1]

Building on these ideas, a number of studies were conducted to test whether the grammatical complexity of a sentence (measured in number of transformations) could be indexed by off-line processing effects (as, for instance, reading or reaction times for the whole sentence).

However, while early work in this direction seemed to support the DTC's assumptions, for instance by showing increased reaction times for passive over active sentences (Miller and McKean, 1964; Savin and Perchonock, 1965; Gough, 1966, a.o.), several studies highlighted fundamental mismatches between the grammar's prediction and experimental data (Slobin, 1966; Fodor et al., 1974; Fodor and Garrett, 1967; Townsend and Bever, 2001, a.o.). This kind of empirical evidence seemed, at the time, to unmistakably prove the fallacy of the DTC, and led many researchers to believe that a theory of parsing which made direct use of the grammar was psychologically implausible, or to formulate alternatives to Chomsky's transformational grammar that were supposed to be more *realistically* realizable.

> [...] Fodor, Bever, and Garrett (1974) conclude that the experimental evidence tends to support the psychological reality of grammatical *structures*, but that the evidence does

[1] This stance differs significantly from Chomsky's views as expressed in some of his other work. For instance:

> [the] generative grammar does not, in itself, prescribe the character or functioning of a perceptual model.
>
> (Chomsky, 1965)

> not consistently support the reality of grammatical *transformations*. [...] In particular, the derivational theory of complexity — the theory that the number of transformations operating in the grammatical derivation of a sentence provides a measure of the psychological complexity in comprehending or producing the sentence — cannot be sustained.
>
> (Bresnan, 1978, pg. 2)

However, there are several reasons why these conclusions are misguided. First of all, the DTC's original predictions were heavily dependent on a particular theory of grammar (Chomsky (1965)'s Standard Theory). Many of the prediction mismatches pointed out by Fodor and Garrett (1967) were later resolved as being problems with the formulation of the grammar or with the original experimental results, and can be reconciled with DTC in a number of ways (Berwick and Weinberg, 1982; Garnham, 1983; Stabler, 1984; Phillips, 1996; Lewis and Phillips, 2015).

Moreover, as Berwick and Weinberg (1983) very clearly argue, the DTC's predictions rely on two particular components: 1) the specification of the grammar (thus, the representations built during processing), and 2) the specification of the parsing system.

In particular, the DTC assumed that each transformational operation was assigned one unit of cost (e.g., in terms of processing time), and that the corresponding parsing operations were to be executed serially (Miller and Chomsky, 1963; Bresnan, 1978). Berwick and Weinberg show that by simply incorporating the grammar in a different parsing system — specifically, a parallel computational architecture — one could easily account for the processing differences reported in the psycholinguistic literature at the time.

At this point, we might wonder what the relevance of this discussion is nowadays, since the notion of *syntactic derivation* has fundamentally changed from Chomsky's early attempts to a transformational grammar (Chomsky, 1995; Hunter, 2019).

Note, however, that Berwick and Weinberg's aim was not to argue for their specific parallel architecture per se, nor for the validity of a particular approach to sentence structure. Instead, their aim was to show that the main problem of the DTC as it was initially understood was in the formulation of its *linking hypothesis* — the connection between grammatical structure and processing behavior — when evaluating psycholinguistics' complexity results, and not in the underlying assumption that such a connection exists.

> In using psycholinguistic experiments to choose between grammars it is not sufficient to present one parser (incorporating some grammar) that can perform a certain task. Rather, one must justify at least in a preliminary way both the grammar and the theory of human computational capacity underlying the parser. More particularly, in order to use psycholinguistic evidence to show that one grammar is more highly valued than another one must provide an independently plausible theory of computational capacity that yields the correct predictions for the experimental data most naturally when coupled with that particular theory of grammar.
>
> (Berwick and Weinberg, 1983, pg. 7)

Importantly, Miller and Chomsky (1963)'s main claim — namely, that mental computations have a cost — is still widely accepted by cognitive psychologists (Phillips, 2003, Chp. 5). Thus, while the specific details of the DTC might not apply directly to the modern state of theoretical linguistics, its fundamental questions remain relevant. A valuable, still unanswered question is then how much of the processing complexity associated with different types of sentences is predicted by the grammatical derivations of modern minimalist syntax.

In order to overcome the original shortcomings of the DTC, it is important to ground the enterprise in a testable, cognitively plausible hypothesis of how structural complexity drives processing behavior. In this sense, this dissertation follows the ideas of Hale (2001) in adopting a framework — based on a weak version of the DTC — in which a computational cost is not associated with single grammar rules directly, but with parsing operations building the surface structure. The cost of one grammatical transformation can thus be spread during the processing phase upon different parsing operations.

In adopting such an approach, we have to ask precise questions about (a) the nature of the structures built during the parsing process, (b) the time-course of the structure building operations connecting linear input to hierarchical representations, and (c) a psychologically reasonable theory of how cognitive resources are linked to parsing operations to derive measures of cognitive load.

Crucially, each of these points can be addressed in several ways, as there are many possible theories of grammar, parsing algorithms, and notions of cognitive load. In this sense, adopting cognitively motivated assumptions is fundamental in building an explanatory theory.

> [...] one must justify at least in a preliminary way both the grammar and the theory of human computational capacity underlying the parser. More particularly, [...] one must

> provide an independently plausible theory of computational capacity that yields the correct predictions for the experimental data most naturally when coupled with that particular theory of grammar.
>
> (Berwick and Weinberg, 1983, pg. 7)

In this dissertation, I follow in the steps of a variety of theories exploring the connection between the time-course of human sentence processing and memory mechanisms. Therefore, the next section presents a discussion of the role played by theories of memory usage in the study of human sentence processing. In Section 2.4, I then detail the specific choices made, for each of these dimensions, by the modeling approach under consideration.

2.3 Memory Limitations in Human Sentence Processing

Memory capacity has a fundamental role in human cognition in general, and in language comprehension specifically.[2] This is particularly relevant in language processing, as comprehenders build sentence representations incrementally[3], but have to keep track of dependencies spanning several phrases or clauses (Marslen-Wilson, 1973; Marslen-Wilson and Tyler, 1980; Tanenhaus et al., 1995; McElree et al., 2003, a.o.). Consider, for instance, the sentence in (7):

(7) Who do the Gems love __?

Here, there is a long-distance dependency between *Who*, perceived at the beginning of the sentence, and the complement position after *love* — where *Who* has to be integrated in order to achieve the correct interpretation (the integration site is also often referred to as the *gap*). Crucially, while the processing system is able to correctly establish that the filler and the gap depend on each other, it

[2]There is as ongoing debate in the psychology literature about whether there are resources exclusively dedicated to the human sentence processing system, or if parsing mechanisms are subject to domain independent cognitive constraints. However, here I do not intend to address these issues. I simply mean to point out that the psycholinguistic literature on working memory evolved somewhat independently from that of other cognitive domains.

[3]Meaning that syntactic representations are built online while comprehending a sentence. It is in fact trivially evident that a person does not wait until the end of the sentence to begin processing. Moreover, *incrementally* in this context also traditionally implies that each word in the input is integrated into the syntactic representation as soon as it is read, an assumption supported by a number of experimental results (Just et al., 2003; Frazier, 1978; Gerth, 2015).

has been demonstrated that the number of words and clauses between the two significantly affects processing time and comprehension accuracy.

Importantly, cognitive scientists have repeatedly pointed out that, even though many cognitive skills often rely on prior perceptual and cognitive analyses, humans have limited resources dedicated to actively attend to old information, while concurrently process new inputs (Anderson, 1996; Broadbent, 2013; Cowan, 2005). Given these limits to our attention abilities, a traditional hypothesis is that the process of tracking the dependency between fillers and gaps is accomplished by storing the former into some kind of working memory — a system that can actively maintain certain amount of information for a short time (Gibson, 1998; Caplan and Waters, 1999; McElree et al., 2003, a.o.).

However, there is good evidence that working memory is itself limited in several ways. In particular, there are limits in *capacity* — as bounds to the amount of information that can be stored at any given time — and *time* — as stored material is slowly forgotten (decays), unless integrated into the syntactic representation (Gibson, 2000; Van Dyke and Lewis, 2003; Lewis and Vasishth, 2005; Just et al., 2003; Nicenboim et al., 2015).

Building on these insights, a variety of sentence processing theories have flourished, that take the comprehension difficulties associated with certain sentences as indexing heavy memory requirements during processing.

2.3.1 Memory-based Approaches to Processing Complexity

Memory-based approaches to sentence comprehension assume, as the name says, that the memory resources available to the human sentence processing system are tightly bound. Importantly, a number of these theories make fundamentally different assumptions not only about the specification of how exactly memory is constrained, but about the *nature* of the memory processes driving the complexity profile of a sentence during processing (Wanner and Maratsos, 1978; Miller and Chomsky, 1963; Rambow and Joshi, 1994; Joshi, 1990; Felser et al., 2017).

Here, I briefly recap two theories representative of the most influential approaches to the study of the memory mechanisms involved in language processing: Dependency Locality Theory (Gibson,

2000) and the activation-based model (Lewis and Vasishth, 2005). While not the oldest among theories suggesting that memory usage affects sentence comprehension in significant ways, these two approaches formulate precise hypotheses that are compatible with the assumptions of modern theoretical linguistics. Moreover, they are supported by a variety of empirical findings, and have inspired a plethora of experimental and computational modeling investigations into the nature of humans' sentence processing mechanisms.

Dependency Locality Theory Dependency Locality Theory (DLT; Gibson, 1998, 2000) hypothesizes two distinct memory components contributing to processing cost: (a) *storage*, the cost of keeping a structure in memory; and (b) *intergration*, the cost of connecting new incoming words into the structure built thus far.

Storage cost is measured by the number of syntactic heads required to complete the current input as a grammatical sentence, and it is thus independent of the amount of time a dependency has to be kept in memory.

On the other hand, integration cost is driven by the need to connect an incoming word to the syntactic representation of the sentence. This includes retrieving the structural representation as built up to that point from memory. Under the assumption that representations (either single words or partial structures) decay in memory over time, structural integration complexity is supposed to increase linearly with the distance between the elements being integrated. Therefore, integration is fundamentally a *locality* based theory of resource commitment.

Importantly, the distance between two elements is not simply identical to the number of intervening words, but instead depends on the number of new *discourse referents* — that is, "an entity that has a spatio-temporal location so that it can later be referred to with an anaphoric expression, such as a pronoun for NPs, or tense on a verb for events" (Gibson, 2000).

Additionally, Gibson has to make several assumptions as to how storage and integration interact. Gibson (1998) postulates that these two elements of memory recruitment access the same pool of resources, and that such resources are bounded up to a fixed quantity. As a result, more resources committed to storage imply slower integration processes, and vice-versa. The complexity of a sentence is then defined as the *maximum* memory load during the parsing of a sentence (as opposed,

for instance, to the average amount of memory consumed).

In sum, the DLT builds on the idea that a sentence' difficulty is indexed by the number of syntactic dependencies that have to be kept track of during processing. However, it focuses only on the number of discourse referents between the dependent and its head and ignores the role of intermediate structure. Importantly though, Gibson mentions that the syntactic complexity of the intermediate integrations most probably plays a role in driving the overall complexity of the sentence (Gibson, 1998; Gerth, 2015). This is exactly the kind of fine-grained details that the model presented in this dissertation aims to address.

The Activation-based Model In the activation-based model (also retrieval-interference theory; Van Dyke and Lewis, 2003; Lewis and Vasishth, 2005; Lewis et al., 2006; Villata et al., 2018), sentence processing is viewed as a series of *cue-based* memory retrieval operations.[4]

Under this approach, words are stored in memory until they can be integrated into the current syntactic structure (*retrieval*). Once again, items in memory are subject to decay, so that the longer a word has to be kept in memory, the more costly the retrieval operation will be. This is very similar to the integration mechanism postulated by the DLT. However, multiple retrievals of the same item can increase its activation level, thus modulating decay in a less straightforward way than the DLT (Van Dyke and Lewis, 2003; Lewis and Vasishth, 2005).

Additionally, this model assumes that linguistic items in memory are represented as feature bundles, subject to *interference effects* due to the features of other linguistic elements. Access to an element in memory is guided by retrieval *cue features*, to be matched with the features of a stored item. This matching operation is made more difficult when there are multiple stored items

[4]Technically, there are two different models assuming that the processing cost associated with the formation of dependencies between non-adjacent words relies on a cue-based retrieval mechanism: the activation-based model (Lewis and Vasishth, 2005) and the direct-access model (McElree et al., 2003; McElree, 2006). These are often used almost interchangeably to refer to retrieval-based theories of memory cost (Nicenboim and Vasishth, 2018), even though they rely on different assumptions about the way memory usage affects reading times and response accuracy. However, while crucial in driving specific processing predictions, these differences between approaches lie mostly in technical details about the way accuracy and latency of the complexity effects are predicted. At a more abstract level however, both models assume that dependency completion relies on a content-addressable cue-based retrieval mechanism that is subject to interference. As in this brief summary I am mostly interested in contrasting models of interference cost to models of storage, I sketch the ideas underlying the activation-based model as a general example of this kind of approaches. The interested reader is referred to Nicenboim and Vasishth (2018, a.o.) for a recent computational comparison of the predictions made by the two models.

with the same set of matching features (*similarity-based interference;* McElree, 2006; Lewis and Vasishth, 2005; Lewis et al., 2006; Jäger et al., 2015). While the activation-based model excludes a direct cost for storage, the role of stored memory elements is captured by interference effects.

2.3.2 The Role of Computational Models

The theories outlined above specify a general framework to associate linguistic input with processing complexity via different notions of memory usage. However, there are several degrees of freedom in how such theories can be operationalized in order to derive precise processing predictions. Incorporating theoretical assumptions into well-specified computational models allows for a level of formal rigor in the exploration of empirical predictions that would be otherwise impossible. In this sense, there are a number of existing models that rely on theories of memory usage to implement structural-based approaches to sentence complexity.

One popular example is the *ACT-R* model, which provides a quantitative framework to test the predictions of the activation-based theory (Lewis et al., 2006, a.o.). This model links memory representations and grammatical knowledge in the form of production rules, and relies on a left-corner parser to simulate the time-course of the structure building operations. The computational implementation stipulates a monotonic relation between reading times and the activation level of the retrieved chunks. Importantly though, the *ACT-R* model fundamentally couches decay and interference rates in domain general cognitive principles. Thus, the contribution of specific sequences of parsing operations to processing difficulty is obfuscated by complex sets of numerical parameters, modulating the magnitude of the complexity effects in non-transparent ways.

A different approach is that of Demberg et al. (2013), who rely on a broad-coverage parser for a psycholinguistically motivated version of Joshi and Schabes (1997)'s Tree Adjoining Grammar (PLTAG; Demberg and Keller, 2008). This model links processing difficulty to parsing complexity directly by formulating a transparent theory of how grammatical information affects the time-course of processing operations, and has been tested against human reading time data from an eye-tracking corpus. Importantly, in this framework *on-line* (word-by-word) complexity is

not computed as a direct function of memory usage exclusively, but integrates decay with the cost of updating predicted syntactic representations (Demberg and Keller, 2009; Demberg et al., 2013). Thus, the contribution of fine-grained grammatical information is obfuscated by how frequency information in the lexicon affects the hypothesis space of the parser.

Finally, Boston (2012) tests a series of constraints argued to account for locality effects in sentence processing. She addresses the competence-performance debate by comparing cognitive constraints to a variety of syntactic constraints, incorporating them in a structurally-rich model sensitive to human cognitive limitations. However, the structural representations postulated by her parsing model are based on dependency grammars, and thus are unable to capture the variety of structural relations generative syntacticians usually care about. In principle though, this approach is the closest to the aims of this dissertation, and I will return to the connections between the two in Chapter 4.

This dissertation follows in the steps of these quantitative models, and investigates syntactic processing from a computationally informed perspective. In particular, in the next section I present a model that formulates a transparent theory of how incrementally building syntactic structure modulates memory usage, in order to derive a set of off-line complexity profiles reported in the psycholinguistic literature. As my aim is to strengthen the bridge between modern syntactic theory and psycholinguistic models of sentence processing, the issue of how syntactic representations are built over time will be explored from the perspective of Minimalist grammars (Stabler, 1996, 2013).

2.4 Minimalist Parsing

Computational models can be essential in overcoming the shortcomings of the oldest version of the DTC, since they allow for an interpretable link between grammatical knowledge and cognitive processes.

This dissertation builds on past research applying computational formalisms to human sentence processing (Joshi, 1990; Rambow and Joshi, 1994; Steedman, 2001; Hale, 2001; Kobele et al., 2013; Gerth, 2015; Graf et al., 2017, a.o.). While different in their technical details and cognitive

commitments, the processing models in these works share a tripartite structure consisting of:

1. a formalized theory of syntax with extensive empirical coverage;
2. a sound and complete parser for the grammatical formalism;
3. a linking theory between grammar and parser in the the form of a complexity metric deriving processing difficulty, which allows for precise psycholinguistic predictions.

Articulating the assumptions behind the psycholinguistic model rigorously allows this kind of approaches to connect grammatical information to performance evidence, while (partially) avoiding the risk of empirical indeterminacy highlighted by Berwick and Weinberg (1983).

This section presents a recent research enterprise (Kobele et al., 2013; Graf et al., 2015b; Gerth, 2015; Graf et al., 2017) that explores these issues by combining a top-down parser (Stabler, 2013) with complexity metrics measuring memory usage, as modulated by the rich structural hypotheses of the most recent version of Chomsky's transformational grammar (Chomsky, 1995; Stabler, 1996).

First, I discuss the intuitions behind the choice of grammatical representations. Then, I present the details of the parsing model and the way the tree traversal strategy affects complexity metrics indexing memory load. In doing so, I review a series of results showing the validity of the approach, in term of coverage for a variety of psycholinguistic phenomena.

2.4.1 Minimalist grammars

Minimalist grammars (MGs; Stabler, 1996, 2011) are a highly lexicalized, mildly context-sensitive formalism incorporating the structurally rich analyses of Minimalist syntax — the most recent version of Chomsky's transformational grammar framework.

Much work has been done in the past on the formal properties of MGs, showing how they can easily accommodate most of the tools of modern generative syntax — for instance, sidewards movement (Graf, 2012, a.o.), copy movement (Kobele, 2006), ATB extraction (Kobele, 2008), extraposition (Hunter and Frank, 2014), and Late Merge (Kobele and Michaelis, 2011; Graf, 2014), among others. As MGs allow for Minimalist analyses to be formalized more or less faithfully, they

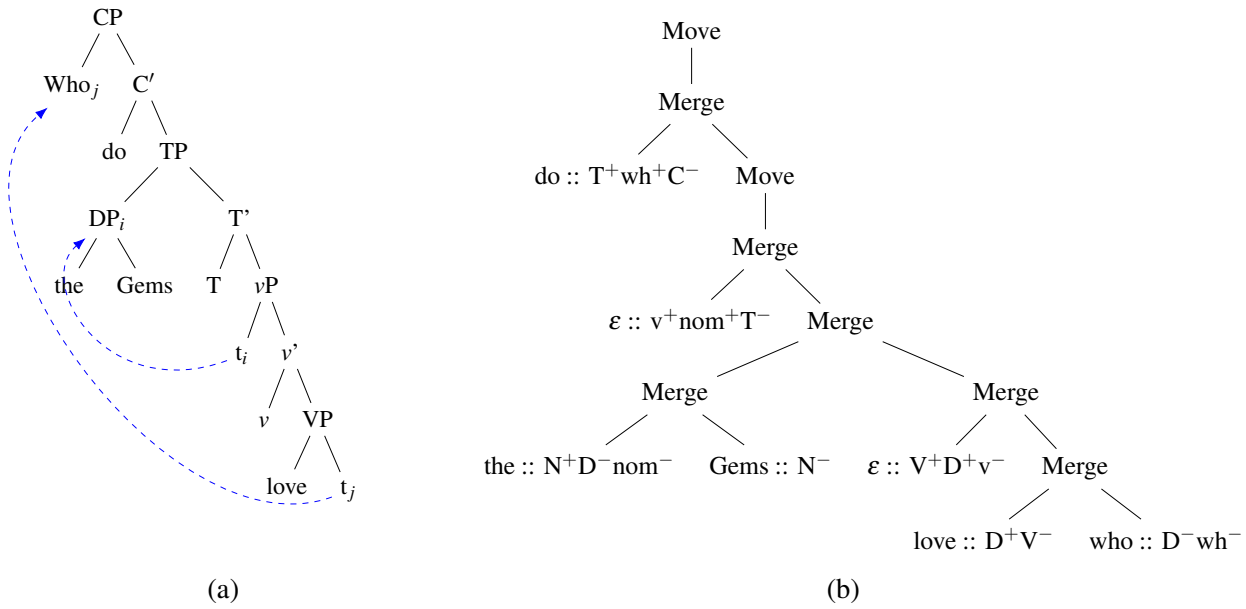


Figure 2.1: Phrase structure tree (a), and MG derivation tree (b), for *Who do the Gems love?*

are particularly suitable towards determing to what extent fine-grained syntactic assumptions can affect processing predictions.

The technical details of the formalism are unnecessary given the focus of this dissertation. I thus introduce MGs in a mostly informal way, as my main goal is to provide the reader with an intuitive understanding of their core data structure: *derivation trees*. The reader is referred to Stabler (2011) or Graf (2013, Chapters 1 & 2; a.o.) for a more formal introduction.

In MGs, a grammar is a set of of lexical items (LIs) consisting of a phonetic form and a finite, non-empty string of features. We distinguish two types of features, each with either *positive* or *negative* polarity: *Merge* features (which I write here in upper caps, with the exception of little *v*), and *Move* features (in lower caps). LIs are assembled via two feature checking operations: *Merge* and *Move*. Intuitively, Merge encodes subcategorization, while Move encodes long-distance movement dependencies.

Informally, Merge combines two LIs if their respective first unchecked features are Merge features of opposite polarity. Move removes a phrase (whose head's first unchecked feature is a negative Move feature) from an already assembled tree and displaces it to a different position (as indicated by a matching positive Move feature; Stabler, 2011). Importantly, a mover always targets

the closest possible landing site (Shortest Movement Contraint; SMC). This constraint makes Move fully deterministic: if there is ever a configuration in which two movers target the same landing site, the derivation is aborted.

Given the SMC, it is always possible to infer which particular sub-tree is to be displaced to which position, exclusively from the feature specifications of the LIs. Thus, Move is represented as a unary branching operation, as there is no need to explicitly specify its arguments (Graf et al., 2017).

MGs succinctly encode the sequence of Merge and Move operations required to build the phrase structure tree for a specific sentence into *derivation trees* (Harkema, 2001; Michaelis, 1998). For instance, Fig. 2.1a and Fig. 2.1b compare these two kinds of trees for the sentence *Who do the Gems love?*. In the derivation tree (Fig. 2.1b), all leaf nodes are labeled by LIs, while unary and binary branching nodes are labeled as Move or Merge, respectively. Crucially, the main difference between the phrase structure tree and the derivation tree is that in the latter, moving phrases remain in their base position, and their landing site can be fully reconstructed via the feature calculus. Thus, the final word order of a sentence is not directly reflected in the order of the leaf nodes in a derivation tree.

As mentioned, derivation trees are the core data structure in MG research. As a record of how a given phrase structure tree is assembled, they contain all the information encoded in the latter. This approach to MGs as generators of derivation trees has numerous technical advantages (Hunter, 2011; Kobele, 2006; Kobele et al., 2007; Graf, 2013). For us though, the main insights come from the perspective that this view provides on parsing: if the structures produced by an MG parser are derivation trees rather than phrase structure trees, MG parsing turns out to be closely related to parsing of context-free grammars (CFGs).

Essentially, MG derivation trees form a regular tree language (Michaelis, 1998; Salvati, 2011, a.o.), and thus — modulo a more complex mapping from trees to strings — can be regarded as a simple variant of CFGs (Thatcher, 1967; Kobele, 2009), which have been studied extensively in the computational parsing literature. This is the core insight behind Stabler's top-down parser.

Before we proceed, a notational clarification. As we will see in the next section, the feature component of the LIs does not play a crucial role in the model used in this dissertation (cf. Chapter

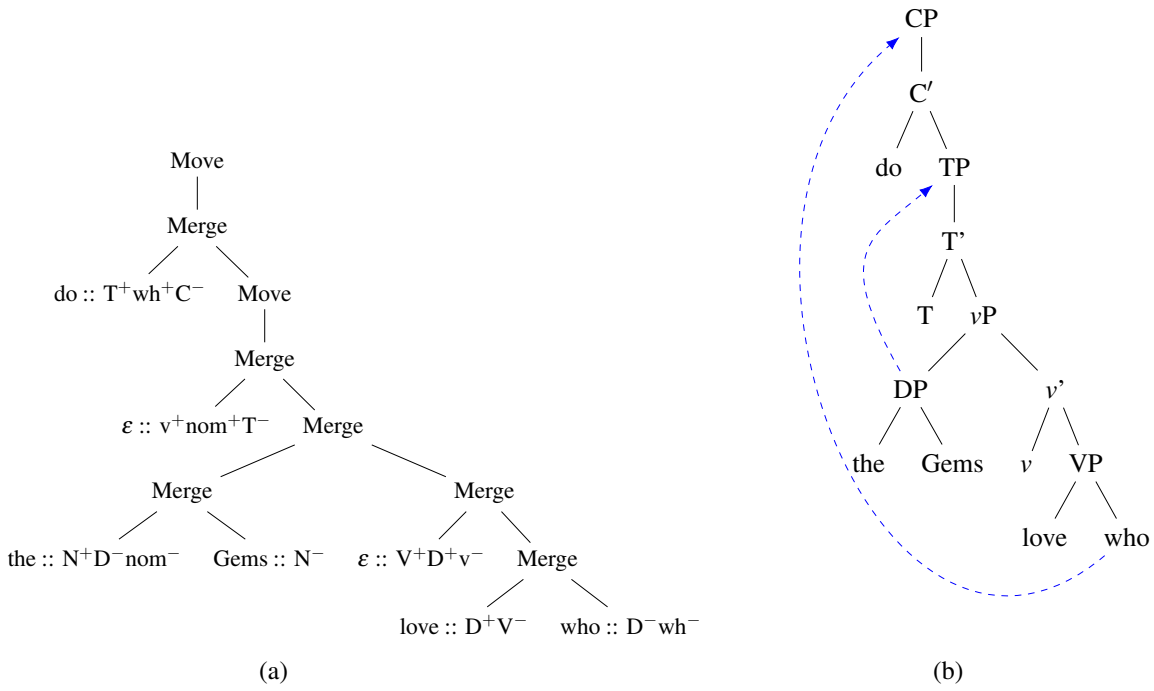


Figure 2.2: Full (a) and simplified (b) MG derivation trees for *Who do the Gems love?*

5). Thus, in what follows I will use a simplified version of derivation trees — in which internal nodes are labelled as in a standard phrase structure tree, features are omitted, and movement elements are linked to their target site with dashed arrows (Figure 2.2b). Moreover, for the sake of clarity, unpronounced LIs are indicated by their category (e.g., C, T, *v*).

2.4.2 Top-down MG Parsing

Stabler (2013)'s parser for MGs is a variant of a standard depth-first, top-down parser for CFGs: it takes as input a sentence represented as a string of words, hypothesizes the structure top-down, verifies that the words in the structure match the input string, and outputs a tree encoding of the sentence structure. Basically, the parser scans the nodes from top to bottom and from left to right; but since the surface order of lexical items in the derivation tree is not the phrase structure tree's surface order, simple left-to-right scanning of the leaf nodes yields the wrong word order. Thus, while scanning the nodes, the MG parser must also keep tracking the derivational operations which affect the linear word order. Without delving too much into technical details, this *string-driven*

Parse Step	Parse Action
step 1	*CP* is conjectured
step 2	*CP* expands to *C'*
step 3	*C'* expands to *do* and *TP*
step 4	*TP* expands to *T'*
step 5	*T'* expands to *T* and *vP*
step 6	*vP* expands to *DP* and *v'*
step 7	*v'* expands to *v* and *VP*
step 8	*VP* expands to *love* and *who*
step 9	***who*** is found
step 10	***do*** is found
step 11	*DP* expands to *the* and *Gems*
step 12	***the*** is found
step 13	***Gems*** is found
step 14	***T*** is found
step 15	***v*** is found
step 16	***love*** is found

Table 2.1: Summary of the actions of a string-driven recursive descent parser for *Who do the Gems love?* as exemplified in Fig. 2.3.

parsing procedure can be outlined slightly more clearly as follows (Kobele et al., 2013):

1. hypothesize the top of structure and add nodes downward (toward words) and left-to-right;

2. if *move* is predicted, it triggers the search for mover $\Rightarrow$ build the shortest path towards predicted mover;

3. once the mover has been found, continue from the point where it was predicted.

An example of this procedure for the sentence *Who do the Gems love?* is given in Fig. 2.3, which illustrates how parser's predictions are matched to the input string. Importantly, note that lexical nodes in each tree are *conjectures* that need to be confirmed against the input. For ease of exposition, the input is enriched with empty lexical heads (C, T, *v*), and a bullet $\bullet$ is used to indicate which lexical item the parser is waiting to integrate into the structure next.

In the example in Fig. 2.3, the parser needs to confirm leaf nodes according to the linear order of items in the input string *Who do the Gems T v love*. Such order then influences which nodes in the tree structure are expanded next. Specifically, since the first word in the input is *who*, the parser does not expand on any left-branching node (e.g., the DP containing *the Gems*) until *who* is found. When that happens, *who* can be *scanned*: the prediction of the parser is matched to the

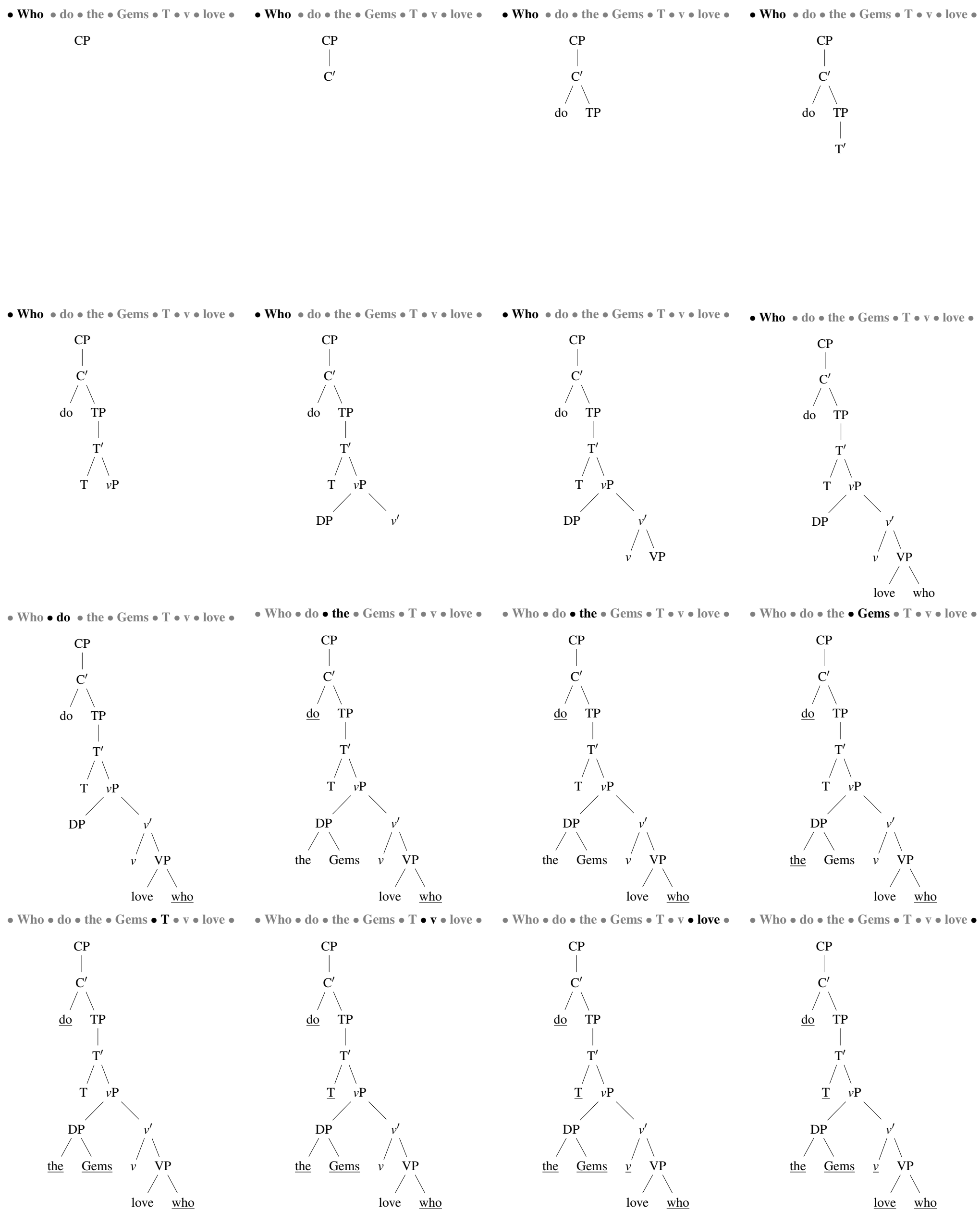


Figure 2.3: Illustrative example of the actions of a string-driven recursive descent parser for *Who do the Gems love?*. For each tree, an *underlined* leaf node is a node that has been both conjectured and confirmed.

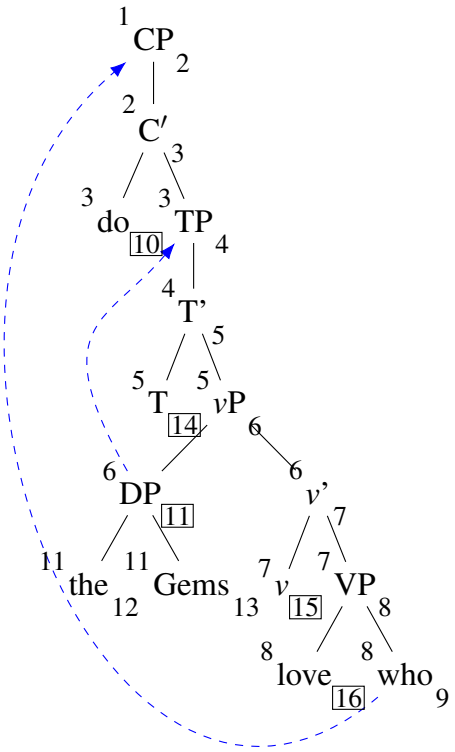


Figure 2.4: Annotated MG derivation tree for *Who do the Gems love?*. Boxed nodes are those with tenure value greater than 2, following (Graf and Marcinek, 2014).

actual input received. Because of this, while *do* is postulated at step 3, it can only be scanned at step 10. Similarly, T can only be scanned after *who*, *do*, and the whole DP *the Gems* have been scanned. A summary of the parser's actions for this example can be found in Table 2.1.

Essential to this procedure is the role of memory: if a node in the tree is hypothesized at step i, but cannot be worked on (scanned) until step j, it must be stored for $j - i$ steps in a priority queue. Moreover, an important advantage of a top-down parser is that the input string is read as a *stream*, and thus we do not require a separate memory buffer to keep hold of it while the structure is being built.

To make the traversal strategy easy to follow, I adopt Kobele et al. (2013)'s notation, in which each node in the tree is annotated with an *index* (superscript) and an *outdex* (subscript). Intuitively, the annotation indicates for each node in the tree when it is first conjectured by the parser (index) and placed in the memory queue, and at what point it is considered completed and flushed from

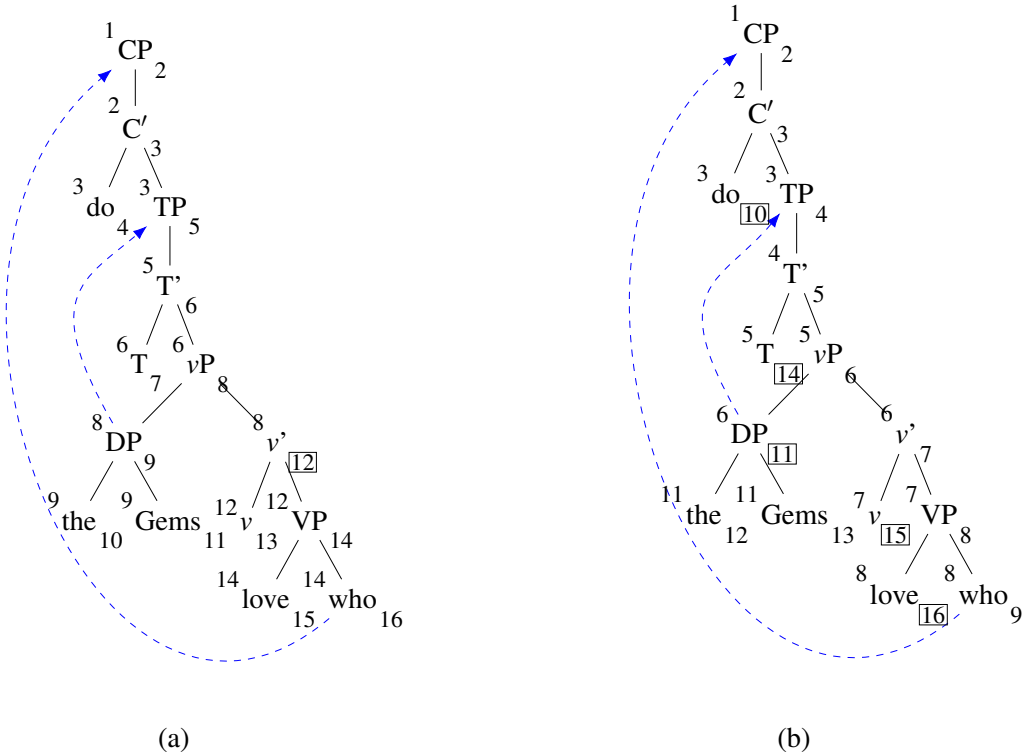


Figure 2.5: Standard recursive-descent tree-traversal (a) compared to the string-driven strategy (b).

memory (outdex). This strategy is shown in Fig. 2.4, which presents an annotated, simplified version of the derivation tree in Fig. 2.1b. The reader is invited to verify how such annotations match the parsing steps in Table 2.1 exactly, and thus allow us to full reconstruct the tree-traversal strategy illustrated in Fig. 2.3.

Importantly, the annotation strategy also clearly highlights in what respect the string-driven nature of Stabler's parser distinguishes it from the tree-traversal strategy of a standard recursive-descent parser.

Consider Fig. 2.5a, illustrating how a standard recursive descent parser would operate over an MG derivation tree for the sentence *Who do the Gems love?*. As discussed before, Move alters the precedence relations between leaf nodes in the derivation tree. But, differently than in a phrase-structure tree, this is *not* reflected by the order of the leaf nodes in the tree itself. In particular, the standard recursive-descent parser tries to scan the leaf nodes in this derivation in the following order: *do T the Gems v love who*. Thus, the parser would first try to reach *do* and scan it. However, since that leaf node does not match the first word in the input (*who*), scanning it would

be impossible, and the parse would be aborted. The problem with the CFG recursive descent parser is in the assumption that the left-to-right order in trees reflects the left-to-right order in the derived string.

Stabler's insight is that the order in which leaf nodes need to be scanned can be inferred from the feature calculus, thus modifying the straight-forward depth-first strategy of the recursive descent. Intuitively, the string-driven recursive descent parser chooses a right branch instead of a left one whenever the right branch contains a mover, and in the input this mover appears to the left of all the material in the left branch (Graf et al., 2017).

In sum, Stabler's top-down algorithm seems to capture some core properties of human language processing strategies: it works incrementally, and it is *predictive* — it makes hypotheses about how to build the upcoming syntactic structure that then need to be confirmed based on the input (Marslen-Wilson and Tyler, 1980; Tanenhaus et al., 1995; Phillips, 2003; Demberg and Keller, 2009, a.o.).

As in other aspects of cognition, prediction also plays a crucial role in language processing. In the MG model, this is reflected by the fact that the predictive abilities of the top-down approach guide how the parser recruits memory resources. However, note that the psycholinguistic literature traditionally refers to prediction in the context of *ambiguity resolution* — the task of choosing between multiple, alternative structural hypotheses available to the parser at specific points during processing — and structural reanalysis (Traxler and Pickering, 1996; Wagers and Phillips, 2009; Chambers et al., 2004; Hale, 2006). These have been shown to have a significant role in determining processing cost (Traxler and Pickering, 1996; Wagers and Phillips, 2009; Chambers et al., 2004), and to be modulated by past experience and generalizations in different ways (Ellis, 2002; Hale, 2006; Levy, 2013).

In this respect, Stabler's original formulation assumes the parser to be equipped with a search beam discarding the most unlikely predictions. Here though, I follow Kobele et al. (2013) in ignoring the beam and assuming that the parser is equipped with a perfect oracle, which always makes the right choices when constructing a tree. Essentially, the MG model considers a deterministic parsing strategy, where ambiguity and reanalysis have no role.

Similarly, human performance during sentence comprehension and production is demonstrably

affected by a variety of factors, including lexical biases (MacDonald et al., 1994), and world knowledge (Chambers et al., 2004; Kentner, 2019), that the MG model purposely ignores. These idealizations are clearly implausible from a psycholinguistic point of view, but were made with a precise purpose in mind: to ignore the cost of choosing among several possible predictions and, by assuming a deterministic parse, to focus on the specific contribution of the grammar to processing difficulty.

2.4.3 Complexity Metrics

In order to allow for psycholinguistic predictions, the behavior of the parser must be related to processing difficulty via a linking theory, which here takes the form of complexity metrics. Specifically, the MG model relies on complexity metrics that predict processing difficulty based on how the geometry of the trees built by the parser affects memory usage.

Based on previous work on MG parsing (Kobele et al., 2013; Graf and Marcinek, 2014; Gerth, 2015), Graf et al. (2017) distinguish three cognitive notions of memory usage: (a) how long a node is kept in memory (*tenure*); (b) how many nodes must be kept in memory (*payload*); (c) how much information is stored in a node (*size*).

Extending the work of Joshi (1990) and Rambow and Joshi (1994), Kobele et al. (2013) assume that one memory unit is allocated per item on the parser's stack, where the stack holds predictions yet to be verified (nodes predicted by the grammar but not scanned yet).[5] Then, *tenure* reflects "*the amount of time that an item is retained in memory*".

Tenure for each node n in the tree can be easily computed via the node annotation scheme of Kobele *et al.*: a node's tenure is equal to the difference between its index and its outdex.[6] Then, the payload of a derivation tree is computed as the number of nodes with a tenure strictly greater than 2.

[5] Following Kobele et al. (2013), I sometimes refer to items being kept in a *memory stack*. Note though that in the actual implementation of the parser the data structure is really a priority queue; cf. (Kobele et al., 2013, Sec 4.1).

[6] Note that Gerth (2015) computes tenure in a slightly different way. For her, the index of a node is not the moment that node was first predicted, but starts from when the features of an item on the stack are processed and scan would be applicable. For instance, if a word is in first position and can be matched against the input immediately, such a word will have a tenure value of 0. Here I follow Graf and Marcinek (2014) and compute tenure consistently with Kobele et al. (2013)'s definition. But note that this difference is not significant in practice, and it is essentially equivalent to ignoring nodes with tenure ≤ 2.

Defining size in an informal way is slightly trickier, as its original conception was based on how information about movers is stored by Stabler's top-down parser (for a technical discussion, see Graf et al., 2015b). Intuitively, size encodes how many nodes in a derivation consume more memory because a certain phrase *m* moves across them. When the top-down parser conjectures a Move node (e.g., CP in Fig. 2.4), it is also conjecturing that it will contain a phrase undergoing movement (thus, a negative movement feature wh^-). To keep track of the movement dependency to be resolved, the parser will carry a wh^+ feature when conjecturing new structure, until a node carrying wh^- is found.

Size, then, provides a way to compute how many additional features the parser has to carry around, increasing memory consumption. Procedurally, the size of the parse item corresponding to each node *n* can be simply computed by exploiting our simplified representation of derivation trees: it corresponds to the number of nodes below *n* that have a movement arrow pointing to somewhere above *n*. In practice, size encodes the additional cost of long movement dependencies over short ones.

Recall now that in this work I am interested in exploring the role grammatical information has in driving *off-line* processing results. However, with the exception of payload, these concepts are not exactly metrics we can use to directly compare derivations. What we are missing is a way for them to be applied to a given derivation as measures of overall processing difficulty. In order to do so, these notions of memory have been used to define a vast set of complexity metrics measuring processing difficulty over a full derivation tree.

2.4.3.1 Base Metrics

Kobele et al. (2013) associate tenure with quantitative values by defining three complexity metrics:

$$\text{MaxT} := max(\{\text{tenure-of}(n)\})$$

$$\text{SumT} := \sum_n \text{tenure-of}(n)$$

$$\text{AvgT} := \frac{\text{SumT}}{\text{BoxT}}$$

MAXT and AVGT[7] measure the maximum and average amount of time any node stays in memory during processing, respectively. In turn, SUMT measures the overall amount of memory usage for all nodes whose tenure is not trivial (i.e., > 2). It thus captures total memory usage over the course of a parse.

Through a series on modeling simulations, Kobele et al. (2013) show that tenure-based metrics (MAXT in particular) can account for the well-established contrasts between center-embedding and right-embedding constructions in English (Bach et al., 1986; Cowper, 1976; Gibson and Thomas, 1999; Miller and Chomsky, 1963), as well as for differences in the processing of nested and crossing dependencies in Dutch and German (Gibson, 2000).

Building on these results, Graf and Marcinek (2014) discuss how MAXT (restricted to pronounced nodes) also correctly accounts for a set of solidly documented processing asymmetries for relative clause (RC) constructions (Goodluck and Tavakolian, 1982; Gibson, 1998, 2000; Gordon et al., 2001; Reali and Christiansen, 2007; O'Grady, 2011; Gibson and Wu, 2013; Frauenfelder et al., 1980; King and Kutas, 1995; Schriefers et al., 1995, a.o.):

- SC/RC $<$ RC/SC

  A sentential complement containing a relative clause is easier to process than a relative clause containing a sentential complement.

- SRC $<$ ORC

  A relative clause containing a subject gap (SRC) is easier to parse than a relative clause containing an object gap (ORC).

However, Graf and Marcinek (2014) point out that, if they consider also the processing contrasts in Kobele et al. (2013), tenure-based metrics find it difficult to account for all of these phenomena at the same time (cf. Gerth, 2015).

Extending Graf and Marcinek (2014)'s analysis of relative clause constructions cross-linguistically, Graf et al. (2015b) also argue for the insufficiency of MAXT as a single,

[7] Consistently with Graf and Marcinek (2014), henceforth I use BOX to distinguish payload as a complexity metric over derivation trees from payload as general concept of memory usage.

Memory Type	Metric
Payload	$\text{BOX} = \lvert\{n \lvert \text{tenure-of}(n) > 2\}\rvert$
Tenure	$\text{MAXT} := max(\{\text{tenure-of}(n)\})$
	$\text{SUMT} := \sum_n \text{tenure-of}(n)$
	$\text{AVGT} := \frac{\text{SUMT}}{\text{BOXT}}$
Size	$\text{MOVERS} := \lvert N \rvert$
	$\text{MAXS} := max(\{\text{size}(n)\})$
	$\text{SUMS} = \sum_{n \in N} i(n) - f(n)$
	$\text{AVGS} := \frac{\text{SUMS}}{\text{MOVERS}}$

Table 2.2: Summary of the base metrics defined in Graf et al. (2017). We refer to n as any node in a derivation tree t. For size-based metrics, N refers to the set of all nodes that are the root of a subtree undergoing movement, while $f(n)$ is the index of the highest Move node the subtree related to the node n is moved to.

reliable metric. Addtionally, Graf et al. (2017) point out that, given the MG parser's sensitivity to fine-grained structural details, a comprehensive evaluation of the model should also modulate processing phenomena over different syntactic analyses.

Based on this observation, Graf et al. (2015b) and Graf et al. (2017) then introduce several new metrics, inspired by those defined for tenure. For example, they define a size-based version of BOX in MOVERS:

$$\text{MOVERS} := \lvert M \rvert$$

where M is the set of all nodes that are the root of a subtree undergoing movement. Essentially, MOVERS encodes the total number of moving subtrees in a given derivation.

They also introduce the equivalent of SUMT for size, which measures the overall cost of maintaining long-distance filler-gap dependencies (O'Grady, 2011). Let N be the set of all nodes of derivation tree t that are the root of a subtree undergoing movement. For each $n \in N$, $i(n)$ is the index of n and $f(n)$ is the index of the highest Move node that n's subtree is moved to. Then SUMS is defined as $\sum_{n \in N} i(n) - f(n)$. The full set of *base* metrics introduced by Graf et al. (2017) is summarized in Table 2.2.

Finally, consider the derivations in Figure 2.6. It should be possible to see how the intermediate

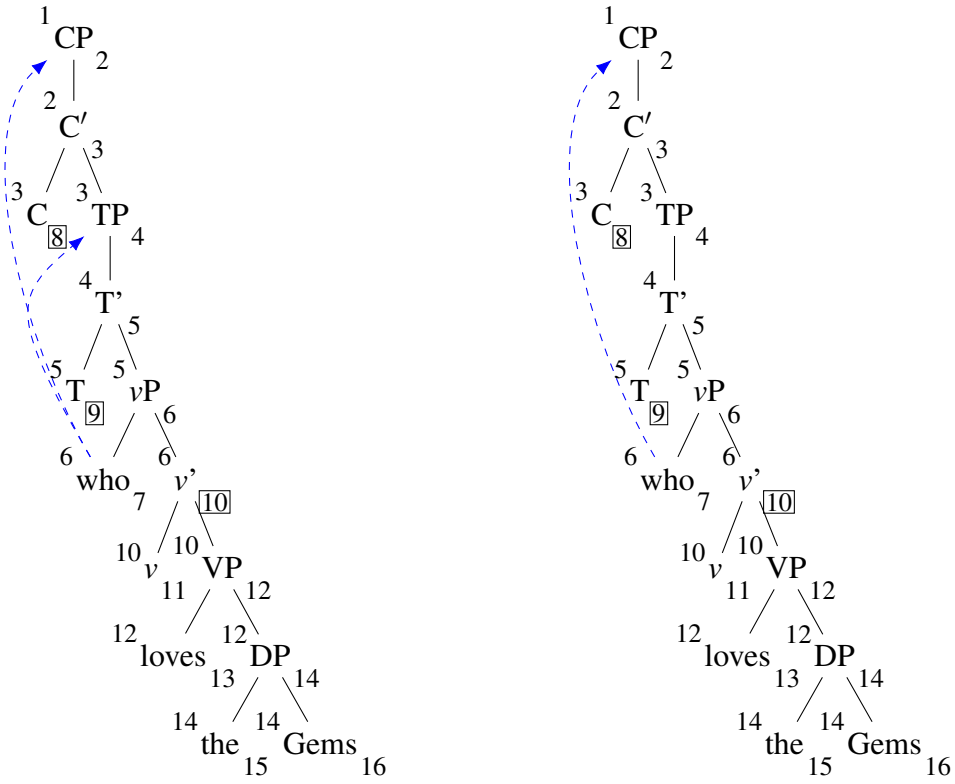


Figure 2.6: Annotated derivation trees for *Who loves the Gems?* with (a) and without (b) intermediate movement steps.

movement step of *who* to Spec,TP on the way to Spec,CP does not change the overall annotation of the tree. This is due to the fact that intermediate landing sites for moved phrases do not affect the traversal strategy (compare Figure 2.6a and Figure 2.6b). Thus, unless otherwise stated, in the following chapters I often do not explicitly highlight them with movement arrows. Because of this, the metrics discussed so far are computed without considering intermediate movement operations.

From a formal perspective, ignoring intermediate landing sites is not an issue. Graf et al. (2015a) prove that any MG with phrases moving to multiple targets can be converted into a strongly equivalent MG, where every phrase moves at most once. However, the question of whether intermediate movement steps matter from a processing perspective is an empirical one (cf. Zhang, 2017). Thus, we must also consider variants of the above metrics that *do* take intermediate movement into account. Consistently with previous work, in what follows I refer to these variants as *prime* metrics: M′.

2.4.3.2 Contrasting Derivations: An Example

As an illustrative example of how these complexity metrics are used to derive off-line processing predictions, consider (8) and (9).

(8) Who __ loves the Gems? **Wh Subject**

(9) Who do the Gems love __ ? **Wh Object**

In (8), there is a dependency between *Who* at the beginning of the sentence and the subject position of *loves*. In (9), this dependency is instead established with the object position. The corresponding derivation trees, annotated by the MG parser with index and outdex values at each node, are shown in Fig. 2.7a and Fig. 2.7b. For simplicity, let us assume that empty heads are also present in the input string, as follow:

(10) Who C T *v* loves the Gems? **Wh Subject**

(11) Who do the Gems T *v* love? **Wh Object**

Consider now the node *C* in Fig. 2.7a. This node is introduced in the memory stack at step 3, as soon as the parser predicts that C' should be expanded in a subtree. However, *C* cannot be scanned

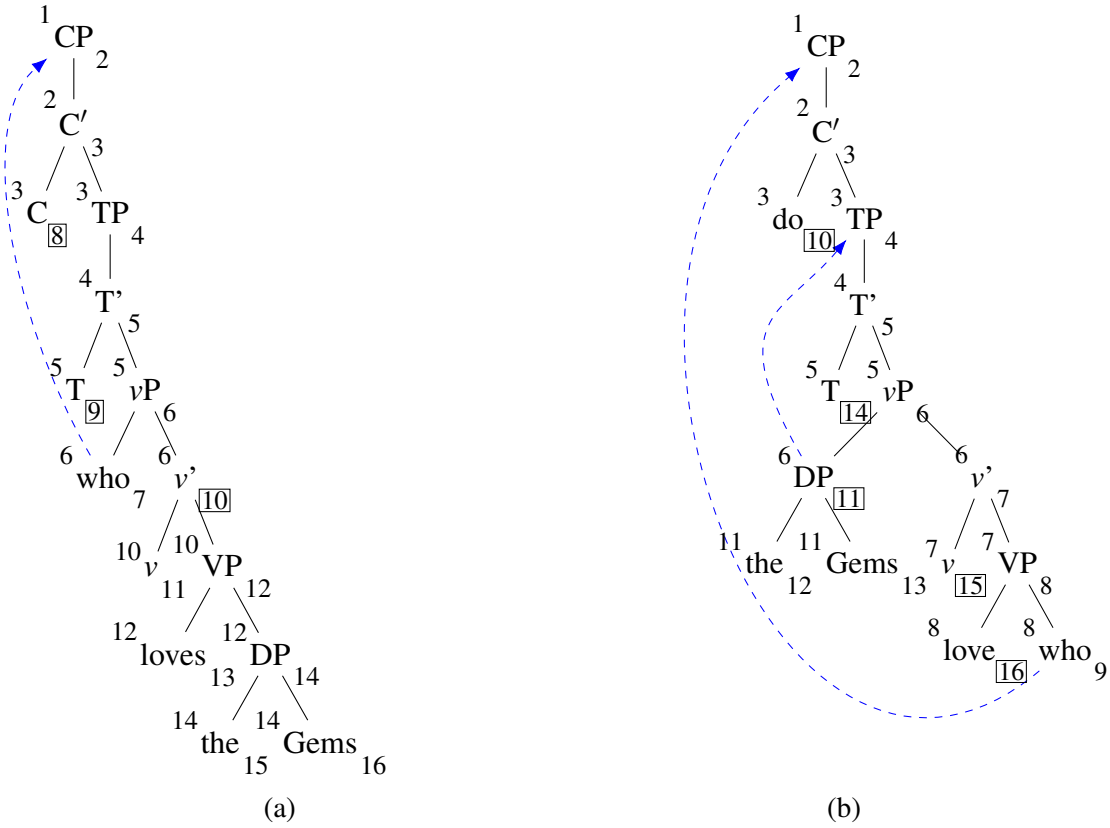


Figure 2.7: Annotated derivation trees for (a) *Who loves the Gems?* and (b) *Who do the Gems love?*

until *who* is found and scanned (at step 6 and 7). Thus, tenure for C is computed as $8-3=5$.

Consider now the same node in Fig. 2.7b (*do*). As before, *do* is introduced at step 3, but it has to wait until the first word in the input (*who*) is scanned. This time however, *who* originates from a lower position, and can only be scanned at step 9. Then, tenure for *does* is $10-3=7$. Note also how movement out of the subject or object position affects the overall traversal strategy. In Fig. 2.7a, once *vP* is expanded the left branch is immediately explored, and the parser goes back to expanding the right-branch of the prediction only after C and T have also been scanned and flushed out of the memory queue. In Fig. 2.7b instead, the left-branch is ignored and the right-branch is expanded so that the parser can explore the quickest path to *who*.

A summary of the non-trivial (i.e., >2) tenure values for the two trees is shown in Table 2.3. As can be seen from this table, the maximum tenure for the derivation in Fig. 2.7a is 5, registered at the node C. Tenure of the derivation in Fig. 2.7b is instead highest at v, where it reaches 8 ($15-7$).

Thus, if we use MAXT to compare the processing difficulty of the two derivations, the model predicts that **Wh Subject** is easier to process than **Wh Object**. While this is essentially a mock example, it seems that the model captures the general psycholinguistic intuition that resolving long movement dependencies is costly.

	Tenure Values				
Wh Subject	***C***	*T*	*v′*		
	$8-3=5$	$9-5=4$	$10-6=4$		
Wh Object	*do*	*T*	*DP*	*v*	*love*
	$10-3=7$	$11-5=6$	$11-16=5$	$15-7=8$	$11-8=3$

Table 2.3: Summary of non-trivial tenure values for the derivations in Fig. 2.7. For each derivation, nodes with MAXT value are bolded and highlighted in red.

This example is also useful in understanding how taking intermediate movement steps into account or not affects the calculation of some metrics. In Fig. 2.7a, *who* moves from its base position in Spec,*v*P to the specifier of CP. However, under standard syntactic assumptions, *who* first needs to move to Spec,TP. As mentioned above, these intermediate steps are not explicitly marked by dashed arrows in the derivation trees. Thus, if we were computing SUMS, we would simply get $6-1=5$, as the metric does not consider intermediate landing sites. However, if we use a variant of this metric that *does* consider intermediate movement, we would have SUMS$' = (6-1)+(6-3) = 8$. For the derivation in Fig. 2.7b, SUMS and SUMS$'$ give us the same result, as both movement operations in this tree occur in one-fell-swoop: SUMS $=$ SUMS$' = (8-1)+(6-3) = 10$.

2.4.3.3 Filters and Recursive Applications

The metrics discussed above are probably the most straightforward in quantifying the three types of memory usages mentioned at the beginning of the section. However, it is possible to refine these metrics in different ways, in order to make them more or less sensitive to different aspect of the structure building process.

For instance, Graf and Marcinek (2014) define a recursive variant of MAXT: MAXT^R. Intuitively, MAXT^R lists the tenure of all nodes in a derivation in descending order. A similar strategy can also be used for size, yielding the complexity metric MAXS^R.

We can then contrast two derivations over MAXT^R, by a pointwise comparison of the two lists. Given two derivations t_1 and t_2, derivation t_1 is easier than derivation t_2 over MAXT^R iff their weights are identical up to position i, at which point t_1's list of MAXT values contains a smaller number than t_2's.

Recursive metrics are interesting in that they allow us to pick up on the contrast between a derivation that has a high MAXT value just on a single node, versus a derivation that has the same identical high value, but on multiple nodes. For instance, take two derivations t_1 and t_2, such that $\text{MAXT}^R(t_1) = [15,5,5,3]$ and $\text{MAXT}^R(t_2) = [15,15,15,15]$. With MaxT, these two derivation trees receive exactly the same score (15), and would thus be predicted to be equally difficult. Instead, MAXT^R predicts t_1 to be easier than t_2.

Apart from recursive evaluations, Graf and Marcinek (2014) suggest that each complexity metric M can be refined by *relativizing* it to specific types of nodes:

M_I refers to M restricted to interior nodes;

M_P refers to M restricted to pronounced leaf nodes;

M_U refers to M restricted to unpronounced leaf nodes.

The split between pronounced and unpronounced leaf nodes has been proposed in the psycholinguistic literature independently of the MG model (Joshi, 1990). Moreover, interior nodes and leaf nodes have a fundamentally different status in syntactic theory. It is then possible to create additional combined variants of the filters above:

M_{IP} refers to M restricted to interior nodes and pronounced leaf nodes;

M_{IU} refers to M restricted to to interior nodes and unpronounced leaf nodes;

M_{PU} refers to M restricted to pronounced and unpronounced leaf nodes (so every leaf node). This metric is referred to M_L in Graf et al. (2017).

M_{IPU} refers to M restricted to interior nodes, pronounced and unpronounced leaf nodes. This is equivalent to the original, unfiltered version of M.

Metric Variants	
M'	M takes intermediate movement steps into account
M^R	applies M recursively
M_I	M restricted to interior nodes
M_L	M restricted to leaf nodes
M_U	M restricted to unpronounced nodes
M_P	M restricted to pronounced nodes

Table 2.4: Variants of the base metrics as discussed in Graf et al. (2017).

Thus, by applying different types of filters, it is possible to derive 6 new metric from each base metric.

2.4.3.4 Ranked Metrics

Lastly, Graf et al. (2015b) introduce the idea of ranked metrics of the type $\langle M_1, M_2, \ldots, M_n \rangle$. These are similar to constraint ranking in Optimality Theory (Prince and Smolensky, 2008): a lower ranked metric matters only if all higher ranked metrics have failed to pick out a unique winner. For instance, take the ranked metric $\langle$MAXT,SUMS $\rangle$. If two constructions result in a *tie* over MAXT, then their SUMS values will decide which of the two is the winner.

Importantly Graf et al. (2017) show that, when complexity metrics are allowed to be ranked in such a way, the total number of possible metrics quickly reaches an astronomical size (up to 1600 distinct metrics). Given such an explosion in the metric space, it would be reasonable to wonder whether such quantitative measure are helping, or if instead are just once again highlighting the empirical difficulties connected to formulating a valid, psychologically plausible linking theory. However, surveying the variety of previously modeled phenomena, Graf et al. (2017) suggest that the number of metrics truly needed to account for human processing contrasts can be reduced to a small number of core metrics. This hypothesis seems supported by recent work on several different constructions cross-linguistically (Zhang, 2017; Liu, 2018; Lee, 2018; De Santo and Shafiei, 2019), and is further explored in Chapter 3 and Chapter 4 of this dissertation.

2.5 Where We Are At, and Where We Are Going

There is no doubt among linguists that the processing ease of a sentence is influenced by some of its underlying structural properties. For instance, an ambiguous prepositional phrase is more easily interpreted as modifying a verb than a noun; and English ORCs are harder than SRCs. However, it is unclear exactly what aspects of grammatical structure are relevant to processing, and to what extent — opening questions about the way theories of grammar can contribute to the study of the cognitive processes underlying sentence comprehension.

> The development of more refined measures of processing load will surely contribute to a more exact understanding of how grammatical information is employed in sentence processing.
>
> (Bresnan, 1978, pg. 57)

This dissertation builds on a specific line of research, which has attempted to look at these issues through the lens of a computational model grounded in a mathematical characterization of modern syntactic theory: Minimalist grammars. Specifically, the MG model relies on a fully specified top-down parser, and set of complexity metrics which link memory usage to the way the parser navigates the geometry of a derivation. The transparent measure of processing complexity offered by these metrics allows for a rigorous study of the precise contribution of grammar to processing.

2.5.1 The MG Model and Its Potential

Looking at the relation between structural complexity and processing effects through the lens of an MG parsing model has been strikingly successful, leading to correct difficulty predictions for several off-line phenomena — such as right-embedding vs. center-embedding, nested dependencies vs. crossing dependencies, as well as a set of cross-linguistic contrasts involving relative clauses. A summary of the variety of such results is presented in Table 2.5.

Among such results, the work of Graf et al. (2017) is particularly interesting, as it presents a detailed replication of past modeling results, modulated across a variety of complexity metrics *and* different syntactic analyses. As the complexity metrics the MG model relies on are

Processing Phenomenon		
right vs center embedding	English	Kobele et al. (2013); Gerth (2015) Graf and Marcinek (2014); Graf et al. (2017)
nested vs. cross-serial dependencies	Dutch and German	Kobele et al. (2013); Graf and Marcinek (2014) Graf et al. (2017)
SC/RC $<$ RC/SC	English	Gerth (2015); Graf and Marcinek (2014); Graf et al. (2017)
SRC $<$ ORC	English	Graf and Marcinek (2014); Gerth (2015) Graf et al. (2015b, 2017)
	Japanese Korean Mandarin Chinese	Graf et al. (2015b, 2017)
SRC $>$ ORC	Mandarin Chinese	Zhang (2017)
Heavy NP shift	English	Liu (2018)
RC attachment ambiguities	English Korean	Lee (2018)
	Persian	De Santo and Shafiei (2019)
Quantifier scope		Pasternak and Graf (2020)

Table 2.5: Summary of past MG processing results.

especially sensitive to changes in syntactic structure, carefully controlling for underlying syntactic assumptions is essential to a full validation of the model.

Furthermore, while past results show that the approach is successful on a variety of off-line processing phenomena, they also highlight how a reliable empirical baseline is crucial. That the reliability of the psycholinguistic data targeted by the parser is important is evident in the contrast between the results in Graf et al. (2017), and those in Zhang (2017). Specifically, Graf et al. (2017) provide a variety of metrics that are able to account for the cross-linguistic preference for subject relative clauses (SRC) over object relative clauses (ORC). However, Zhang (2017) argues that the correct preference for Mandarin Chinese is the inverse one (ORC $<$ SRC), and that the number of metrics that are able to capture this contrast *together* with the SRC $<$ ORC preference in other languages is significantly reduced.

Moreover, Zhang (2017) points out that the existing MG metrics are unable to reproduce the complexity profiles she found for the processing of stacked relative clauses in English and Mandarin Chinese. She argues that this failure is due to a fundamental limitation of the current definition of the model: the inability to account for how features drive a derivation. These considerations are also in line with Kobele et al. (2013)'s observation that the MG model's notion of memory is extremely general, and somewhat detached from current cognitive assumptions about

human memory.

Relatedly, I mentioned how the metrics discussed above were motivated by psycholinguistic assumptions about the nature of human working memory (O'Grady, 2011; Rambow and Joshi, 1994; Wanner and Maratsos, 1978; Gibson, 1998, a.o.). However, one could reasonably wonder why we should rely on these specific metrics and not others.

Clearly, we can easily conceive of metrics that take syntactic information into account in different ways (Yngve, 1960; Wanner and Maratsos, 1978; Rizzi, 1990; Rambow and Joshi, 1994; Gibson, 2000; McElree et al., 2003; Lewis and Vasishth, 2005). However, tenure, payload, and size exclusively refer to the geometry of a derivation tree without additional assumptions about the nature of its nodes. Thus, they arguably rely on the simplest possible connection between memory, structure, and parsing behavior. Importantly, Zhang (2017)'s results can be the starting point of a discussion of whether/how psycholinguistic insights can be used to refine the MG model's approach to memory usage. Addressing these issues will be the purpose of Chapter 5.

2.5.2 In Defense of Idealization

As mentioned multiple times already, computational models have a variety of degrees of freedom, which become critical in developing cognitively plausible generalizations. Therefore, before moving forward, it is worth assessing once again the assumptions that the MG model makes with respect to some of the most crucial aspects of a psychologically grounded theory of processing.

First of all, the model relies on a top-down parsing algorithm. One of the most interesting aspects of top-down parsers is that they are purely predictive: the input string is only checked against fully built branches — those that end in a terminal symbol. We already discussed how the human parser is strongly predictive, so this aspect of the top-down strategy seems to be highly desirable. However, the prediction process of the human parser differs from pure top-down parsing, as it seems to be actively guided by the input — and in fact shows evidence of *bottom-up* decisions. Because of these considerations, it has often been claimed that *left-corner* parsing algorithms might be a better fit for the sentence processing strategies employed by humans (Resnik, 1992, a.o.).

As left-corner parsers for MGs now exist (Stanojević and Stabler, 2018; Hunter et al., 2019), the

reader might reasonably wonder what is the value of a modeling approach that explicitly relies on a top-down strategy. In this sense though, it is important to note that — since we don't have access to the direct implementation of the parser in the human mind — the only evidence we have is that human processing behavior seems to integrate top-down and bottom-up elements. Left-corner parsing algorithms are just one possible way to accomplish this.

From a formal perspective, left-corner behavior can be replicated by a top-down parser operating on the left-corner transform of the grammar; or by integrating bottom-up filtering on its search space (Rosenkrantz and Lewis, 1970; Aho and Ullman, 1973; Johnson, 1996; Sikkel, 2012). Then, the overall tree-traversal strategy of the MG parsing algorithm introduced earlier in this chapter would not be affected. Additionally, it is unclear whether the existing implementations of left-corner parsers for MGs actually reflect the desired properties of the human processing system (cf. Hunter, 2018b). Clearly, whether any of these algorithms would give good results when compared to human performance is an empirical question, which needs further investigation. However, in this dissertation I focus on a top-down parser in the attempt to isolate the contribution of the predictive component of the parser to processing complexity.

Secondly, the linking assumption between grammar and parser behind the MG model relies on metrics measuring processing difficulty as indexed by memory load. There is, of course, also a plethora of alternative measures of complexity that do not rely directly on memory burden (Hale, 2001).

In particular, in this chapter I chose to put aside a series of theories of cognitive load — known as *expectation-based* theories — that adopt a view of parsing difficulty grounded in probability. For instance, Hale (2006) assumes that comprehension difficulties associated with specific syntactic structures can be modeled by word-by-word probabilistic measures (surprisal, entropy; Yun et al., 2015; Hale, 2006, 2016), derived from a probabilistic phrase structure grammar. In a sense, this approach follows directly into the steps of Berwick and Weinberg (1983)'s parallel architecture proposal, as cognitive load is measured as a function of the total probability of the structural options that have been disconfirmed at some point in a sentence (Hale, 2011, 2006, a.o.).

While there is no doubt that probabilistic information plays a role in parsing (Ellis, 2002), this dissertation's focus on memory-burden models is motivated by the desire to explore the cost of

structural complexity independently from other factors contributing to processing difficulty. This is also consistent with the decision to discard the beam-search component of Stabler's parsing algorithm in favor of a deterministic approach. Importantly though, the top-down MG parser is compatible with a probabilistic view of grammar, and in principle it allows for the integration of memory-burden and expectation-based theories of processing complexity (Hunter and Dyer, 2013; Gerth, 2015).

Finally, the choice of parsing strategy and memory metrics also affects the type of experimental data we consider. As mentioned, this dissertation is concerned with modeling complexity profiles as reflected by *off-line* processing effects. In psycholinguistics, off-line tasks are concerned with the overall complexity of a sentence. While such tasks provide useful information about processing effects, recently the psycholinguistic literature has become more and more concerned with *on-line* effects — reflecting real-time, word-by-word aspects of the process of sentence comprehension (Clifton et al., 1994; Clifton, 2015).

Notably, when studying the fine-grained time-course of sentence complexity, it seems reasonable to expect factors such as ambiguity and lexical probability to play a decisive role. Similarly, ignoring the bottom-up component of the parser might have an impact on where the locus of complexity is found. As here we are making the explicit choice to ignore the contribution of these elements to processing complexity, for the moment it seems preferable to restrict our investigations to off-line effects. For similar reasons, the MG model does not attempt to predict the magnitude of such effects, and instead replicates processing asymmetries categorically.

Importantly, complexity metrics for MG derivations have recently been used as predictors of on-line, word-by-word complexity (Gerth, 2015; Brennan et al., 2016, a.o.). For instance, Brennan et al. (2016) use a *node-count* metric, which is very similar to tenure as defined here. However, *node-count* is less committed to the structural representation considered (e.g., derivation trees vs. phrase structure trees), and thus it is actually less sensitive than tenure to the specificity of the tree-traversal strategy. A more interesting approach is explored by Gerth (2015), who attempts to combine structure-driven memory metrics with information-theoretical ones. While beyond the scope of this dissertation, Gerth (2015)'s suggestions and encouraging results could be used in the future to define plausible on-line MG metrics from the current off-line ones.

In sum, the model adopted in this dissertation makes several idealized assumptions about the parsing system: it relies on a top-down, deterministic parsing strategy to predict off-line processing complexity exclusively, via metrics that are only sensitive to information about the geometry of a derivation tree. Clearly, if the aim is to build a comprehensive model of human sentence processing, future work will have to relax such assumptions and explore other factors contributing to processing difficulty. However, these idealizations give us a maximally simple model that relies predominantly on syntactic structure, which already has many important insights to offer. Showing this is the goal of the upcoming chapters, in which I discuss Italian Relative Clauses (Chapter 3), gradience in acceptability judgments (Chapter 4), and syntactic priming effects (Chapter 5).

Chapter 3

A Case Study: Italian Relative Clause Asymmetries

3.1 Introduction

As discussed in the previous chapter, a top-down parser for Minimalist grammars (MGs; Stabler, 1996, 2013) successfully models sentence processing preferences across a variety of phenomena cross-linguistically (Kobele et al., 2013; Gerth, 2015; Graf et al., 2017, a.o.). This is done by formulating a transparent theory of how parsing behavior relates to memory usage, thus connecting longstanding ideas about memory engagement during off-line processing with explicit syntactic analyses in rigorous ways. However, while the variety of constructions modeled so far in the literature is encouraging, extending the range of phenomena that the parser correctly accounts for is still crucial to confirm the empirical feasibility of the approach.

In this chapter, I test the MG parser's performance on the processing asymmetries reported for Italian relative clauses, which have been object of extensive study in the psycholinguistic literature. Apart from conforming to a well-attested cross-linguistic preference for subject over object relatives, Italian speakers also show increased processing difficulties when encountering relative clauses with subjects in postverbal position. This difficulty gradient has often been accounted for in the literature in terms of the cost of local ambiguity resolution. Since in the

particular formulation of Kobele et al. (2013) the MG parser acts as an oracle and deliberately ignores structural ambiguity, these constructions thus make for a challenging testing ground for a model attempting to account for processing contrasts *just* in terms of *structural complexity*.

3.2 Modeling Italian RCs

This section reviews the psycholinguistics literature on the processing of Italian postverbal subject, and presents the test cases modeled in the rest of the chapter. As in the MG approach memory usage is modulated by subtle structural differences, I also discuss several modeling choices that had to be made with respect to the syntactic analysis for these constructions. A detailed analysis of the modeling results is then the focus of Section 3.3, which shows how the MG parser succeeds in predicting the correct processing preferences.

3.2.1 Processing Asymmetries

Restrictive relative clauses (RCs) in Italian have been the focus of extensive experimental studies from the perspective of comprehension (Volpato and Adani, 2009), production (Belletti and Contemori, 2009), and acquisition (Volpato, 2010; Friedmann et al., 2009). Italian speakers conform to the general cross-linguistic preference for subject over object RCs (Frauenfelder et al., 1980; King and Kutas, 1995; Schriefers et al., 1995, a.o.), so that (12) is easier to process than (13):

(12) Il cavallo che ha inseguito i leoni
The horse-SG.M that has-SG.M chased the lions-PL.M
"The horse that chased the lions" **SRC**

(13) Il cavallo che i leoni hanno inseguito
The horse-SG.M that the lions-PL.M have-PL.M chased
"The horse that the lions chased" **ORC**

Interestingly, Italian also allows for sentences like (14), ambiguous between a SRC interpretation (14a) and an ORC interpretation (14b) with the embedded subject expressed postverbally (ORCp):

(14) Il cavallo che ha inseguito il leone
The horse-SG.M that has-SG.M chased the lion-SG.M

a. "The horse that chased the lion" **SRC**

b. "The horse that the lion chased" **ORCp**

Although postverbal subject constructions are very common in Italian, in such cases native speakers show a marked preference for the SRC interpretation over the ORCp one. As Italian is a morphologically rich language, sentences like (14) can also be disambiguated by grammatical cues like subject-verb agreement:

(15) Il cavallo che hanno inseguito i leoni
The horse-SG.M that have-PL.M chased the lions-PL.M
"The horse that the lions chased" **ORCp**

In (15), the DP *i leoni* is plural, while the DP *il cavallo* is singular. As in Italian the verb agrees in number with its subject, and in this case the embedded verb is marked for plurality, (15) can only be interpreted as an ORCp construction. However, even in these unambiguous cases studies report increased efforts with ORCps, leading to the following difficulty gradient: SRC $<$ ORC $<$ ORCp (Utzeri, 2007, a.o.).

The contrast between SRCs and ORCs has been well studied in the past, and it is compatible with a variety of processing difficulty models, such as surprisal (Levy, 2013), cue-based memory retrieval (Lewis and Vasishth, 2005), the active filler strategy (Frazier, 1987), the Dependency Locality Theory (Gibson, 1998, 2000), the Competition Model (Bates and MacWhinney, 1987), the Minimal Chain Principle (De Vincenzi, 1991), among many. The increased complexity reported for ORCs with postverbal subjects is more of a challenge to some of these models (e.g., for the Competition model and Dependency Locality Theory; Arosio et al., 2009), as the gap between the moved object and its base position is identical in both configurations. However, this processing profile can be explained in terms of economy of gap prediction and cost of structural re-analysis, due to the possible ambiguity in ORCps at the embedded subject site — where the parser has the choice of either postulating a null pronominal subject or establishing a filler-gap dependency. Crucially though, such an account has to come with extra assumptions about *why*, when building these dependencies, it is preferable to choose one strategy over the other.

Importantly, the aim of this chapter is not to argue for the correctness (or lack thereof) of these accounts. My purpose is to extend previous evaluations of memory metrics for a top-down MG parser as a reliable model of processing difficulty. A successful outcome might also give us insights into how memory-based heuristics can be integrated in more standard filled-gap approaches to on-line processing.

As discussed above, the MG parser has already been successful in accounting for RC asymmetries cross-linguistically (Graf et al., 2017; Zhang, 2017). Thus, Italian RCs are the perfect next step in understanding the plausibility of the model, allowing us to build on the insights provided by previous work while incrementally exploring new structural configurations. In particular, the fact that, by assumption, the MG parser ignores structural ambiguity (thus potential costs associated to re-analysis) and deterministically builds only the correct parse, makes ORCs with postverbal subjects an intriguing test case.

3.2.2 Syntactic Assumptions

The central tenant of the MG model is to take syntactic commitments seriously, so to explore how different aspects of sentence structure drive processing cost. The choice of a syntactic analysis is then particularly important. In line with most of the psycholinguistic literature on Italian RCs (Arosio et al., 2017, a.o.), I adopt a promotion analysis of relative clauses (Kayne, 1994).

A sketch of this analysis is shown in Fig. 5.1. The head noun starts out as an argument of the embedded verb and undergoes movement into the specifier of the RC. The RC itself is treated as an NP, and selected by the determiner that would normally select the head noun in more traditional, head-external accounts (Chomsky, 1977).

Furthermore, again consistently with the Italian psycholinguistic literature, my analysis of postverbal subjects follows Belletti and Leonini (2004, a.o.). Consider the following declarative clause with a postverbal subject:

(16) Inseguono il cavallo i leoni
Chase the horse the lions
"The lions chase the horse"

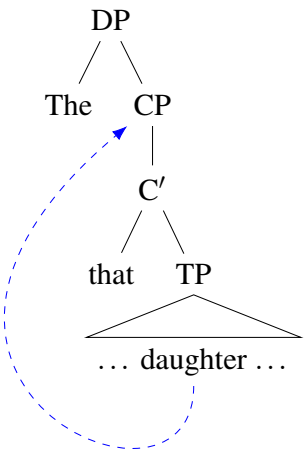


Figure 3.1: A sketch of Kayne's promotion analysis for the relative clause [$_{DP}$ The [$_{CP}$ daughter$_i$ [that t_i was on the balcony]]].

According to Belletti and Leonini, in postverbal constructions the subject DP (*[i leoni]*) is merged in preverbal subject position Spec,*v*P, and then raised to a Spec,FocP position in the clause-internal *v*P periphery. The whole verbal cluster is raised to a clause-internal Spec,TopP position; and an expletive *pro* is base generated in Spec,TP and co-indexed with the postverbal subject (Fig. 3.2).[1]

3.3 Modeling Results

With all preliminaries in place, we can now move to modeling the Italian processing asymmetries with the MG parser, following the approach detailed in Chapter 2. In particular, derivations for each test sentence are fed to the parser, together with the processing contrasts reported by the psycholinguistic literature — reframed in terms of pairwise comparisons (e.g., $SRC < ORC$, $ORC < ORCp$, etc.). In order to derive processing predictions, the parser is then equipped with

[1] Technically, Belletti and Leonini (2004) assume that VP, not *v*P, raises to Spec,TopP. This follows from the authors adopting Collins (2005)'s smuggling analysis of passives directly. However, if we follow the traditional view of active verbs moving out of their base position to adjoin to little *v*, this analysis cannot hold as it would derive the wrong word order. Thus, I raise the whole *v*P cluster to TopP. This also seems to be in the spirit of what suggested by Belletti and Contemori (2009). But note that the modeling results in the following section would remain mostly unchanged even if we were to leave the *v*P shell in its base position, while both verb and object raise above.

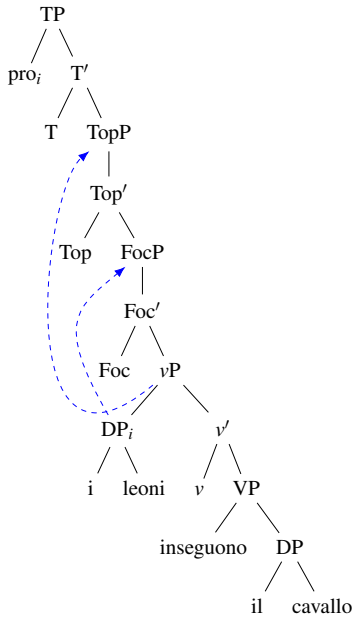


Figure 3.2: Belletti & Leonini's analysis for the sentence in (16).

the complexity metrics defined by Graf et al. (2017), and reviewed in Chapter 2.

3.3.1 Core Results

For consistency with psycholinguistic stimuli, and with previous MG parsing work, RCs are not modeled by themselves, but are embedded in a template sentence. Thus, I first tested the parser performance on *right-branching* restrictive RCs of the form *(pro) vedo il cavallo [$_{RC}$ che ...]* (*I see the horse [$_{RC}$ that ...]*) — the RC head raising to the matrix *object* position, and the relative clause either an SRC (12), an ORC (13), or an ORCp (15).

The corresponding derivation trees, annotated by the MG parser with index and outdex values at each node, are shown in Fig. 3.3a, Fig. 3.3b, and Fig. 3.3c respectively. Recall that by assumption the parser is equipped with a perfect oracle, and that the complexity metrics are *only* sensitive to structural differences (i.e., the MG model is blind to agreement relationships). Contrasting (12) and (15) is then equivalent to contrasting (14a) and (14b). Thus, to reiterate the central tenants of the approach, these comparisons aim to model both the preference for SRC in ambiguous cases, and the overall increased processing difficulty of ORCps, just in terms of structural differences.

Modeling results show that the parser correctly predicts the gradient of difficulty observed for

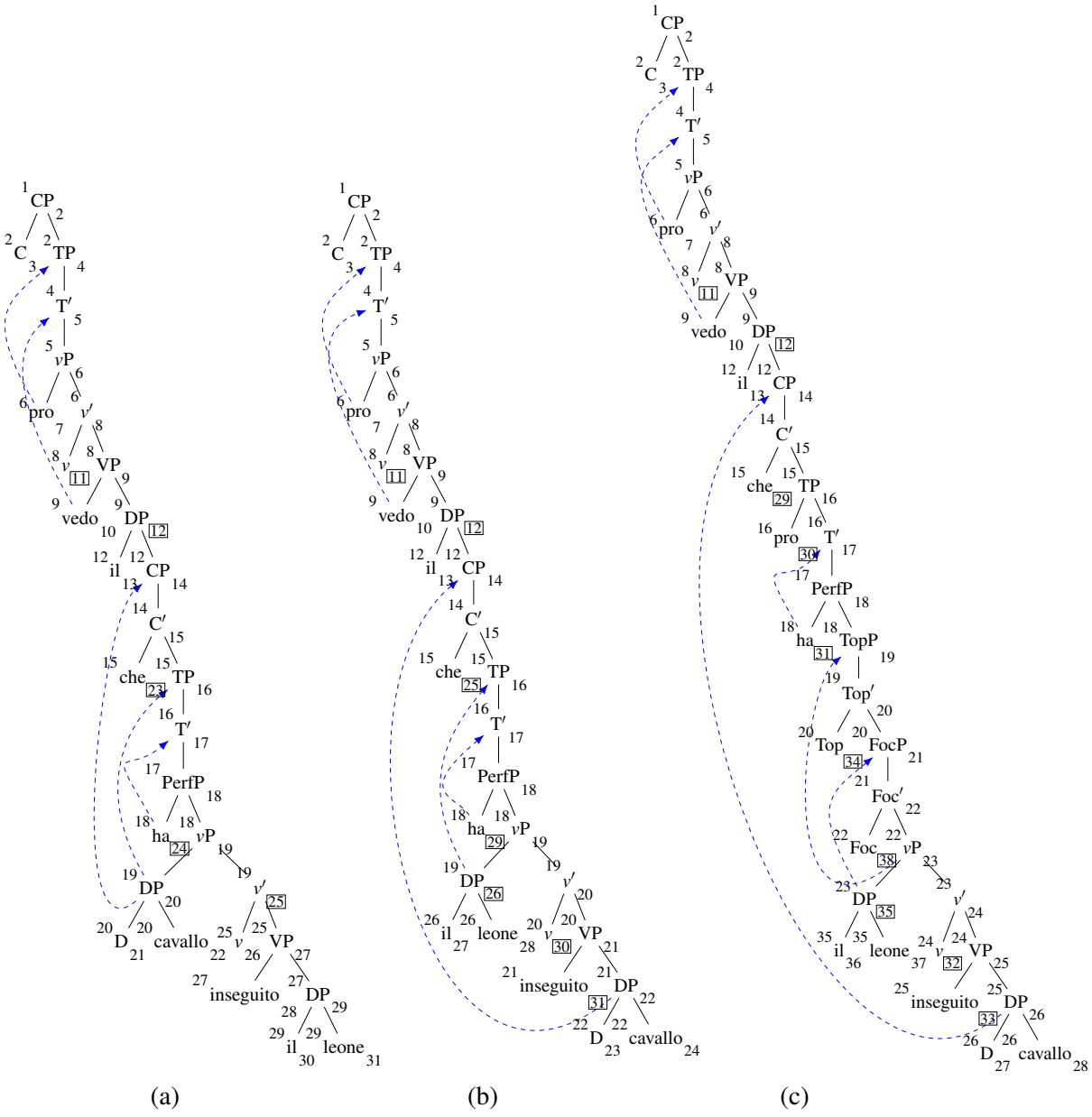


Figure 3.3: Annotated derivation trees for right-embedding (a) SRC, (b) ORC, and (c) ORCp.

Italian RCs (SRC < ORC < ORCp), across a variety of complexity metrics.[2] In fact, the increased difficulty of ORCps over both SRCs and ORCs is predicted by *every* base (i.e., non ranked) metric defined by Graf et al. (2017). A complete summary of how each metric fares on the Italian asymmetries can be found in Appendix A. However, since the relationship between complexity metrics and the structure of a specific derivation tree is subtle, in what follows I focus exclusively on two metrics that have been noted in previous studies as consistent predictors of processing difficulty: MAXT and SUMS (see Table 3.1).

Clause Type	**MaxT**	**SumS**
obj. SRC	8/*che*	18
obj. ORC	11/*ha*	24
obj. ORCp	16/*Foc*	31

Table 3.1: Summary of MAXT (*value*/*node*) and SUMS by construction, for the right-embedding RCs in Fig. 3.3. Obj. indicates the landing site of the RC head in the matrix clause. The expected difficulty gradient is SRC < ORC < ORCp.

The fact that MAXT (SRC: 8/*che*; ORC: 11/*ha*; ORCp:16/*Foc*) succeeds in predicting the reported processing preferences is encouraging, given the past success of this metric on many different constructions.[3] In particular, observe how the string-driven traversal strategy of the MG parser makes tenure sensitive to minor structural differences. In the SRC, *che* is introduced at step 15. Since, based on information in the input string, the parser is looking for the the subject DP *il cavallo*, *che* has to be kept in memory until the latter is found. Thus, it is flushed from memory at step 23. In the ORC, *che* is also put in memory at step 15. However, since the head of the relative clause is the embedded object, the parser will discard the standard CFG top-down strategy, ignore the subject DP, and keep expanding nodes until *il cavallo* is found. Thus, *che* cannot be flushed from memory until step 25.

The difference between SRC and ORC also highlights how tenure interacts with movement. Once *che* has been found in the SRC tree, the next node in the stack is *ha*, which can be discharged from memory immediately after. In the ORC however, the parser still has to find the subject DP.

[2]Code for all the simulations in this chapter is available at https://github.com/CompLab-StonyBrook/mgproc.

[3]These predictions hold even if we ignore tenure on unpronounced nodes — as suggested by Graf et al. (2017) — since we would obtain (SRC: 8/*that*; ORC: 11/*has*; ORCp:14/*that*).

Thus, *ha* has to be kept in memory for the three additional steps that are required to conjecture and scan *il leone*. Similarly, the maximum tenure recorded on the Foc head in ORCp highlights the cost of the additional movement steps postulated for this construction. The Foc node needs to wait until both the RC object *and* subject are built and scanned, before being itself discharged from the memory queue.

3.3.2 Additional Simulations

From one side, the successful predictions made by MAXT are a welcome result, as they confirm the sensitivity of tenure-based metrics to fine-grained structural details. From the other though, one might wonder exactly how much these differences depend on the specific case study we are modeling. In this section, I partially address this issue by looking at variations in the construction of the RCs, and at two more processing asymmetries involving Italian post-verbal subjects. I return to the general issue of the sensitivity of the MG results to syntactic choices in Sec. 4.5.

3.3.2.1 Left-Embedding RCs

Due to the string-driven nature of its traversal strategy, the MG parser is peculiarly sensitive to the depth of left- vs. right-embedding constructions. To control for this, I tested the parser predictions on sentences of the form *Il cavallo [$_{RC}$ che ...] salta la siepe* (*The horse [$_{RC}$ that ...] jumps the fence*, Fig. 3.4), with the head noun raising to the *subject* position in the matrix clause.

Clause Type	**MaxT**	**SumS**
subj. SRC	21/*v*'	37
subj. ORC	21/*v*'	44
subj. ORCp	28/*v*'	56

Table 3.2: Summary of MAXT (*value*/*node*) and SUMS by construction, for the left-embedding RCs in Fig. 3.4. Subj. indicates the landing site of the RC head in the matrix clause. The expected difficulty gradient is again SRC $<$ ORC $<$ ORCp.

In this new context, MAXT predicts that SRC and ORC should have the same processing complexity (they *tie*), since their memory differences are flattened by the increased tenure on the matrix *v'* (the Merge node expanding the matrix *v*P). The tenure of this node depends on the size

of the matrix subject — thus, on the size of the relative clause. Since the size of the SRC and of the ORC is the same (the only thing changing being the site of extraction), MAXT for the whole sentence will never vary between the two constructions. This issue is solved by SUMS, which correctly predicts SRC < ORC, as well as the SRC/ORC < ORCp contrast (see Table 3.2).

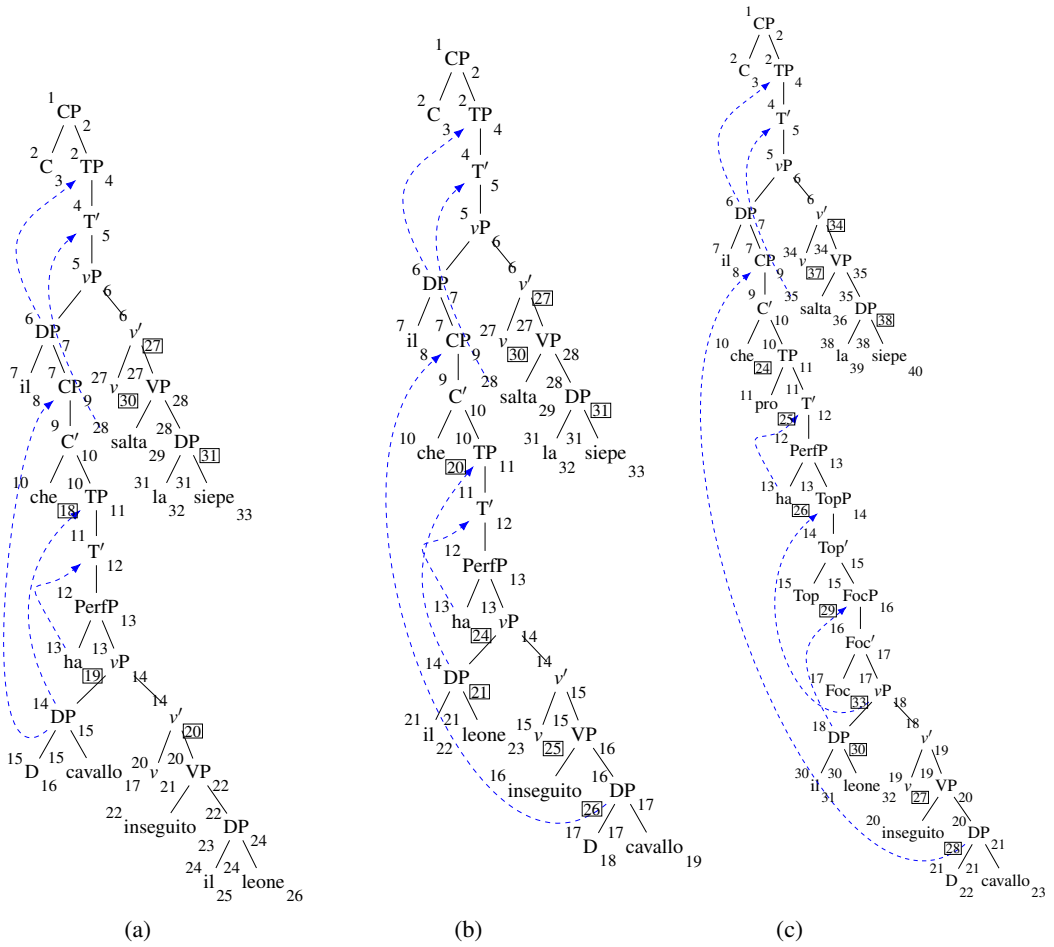


Figure 3.4: Annotated derivation trees for left-embedding (a) SRC (b) ORC and (c) ORCp.

Interestingly, MAXT also correctly predicts the increased difficulty of ORCps in these left-embedding cases. As seen above, MAXT flattens the differences in clauses with subject-modifying SRC/ORCs because the size of the RCs in subject position is identical. This is not the case for ORCps, due to the sequence of projections and movement steps involved in deriving postverbal subjects from the base SVO order. Thus, while MAXT in these sentences is still measured on the matrix *v'* (28), this value is also picking up on the additional steps required

to derive the internal structure of the ORCp construction.

3.3.2.2 Postverbal Subjects in Matrix Clauses

In order to understand the complexity of the grammatical assumptions made for the postverbal subjects, we can look at processing asymmetries of postverbal constructions outside of RC environments. Consider Italian declarative sentences like in (17).

(17) Ha chiamato Gio
Has called Gio
a. "He/she/it called Gio" **SVO**
b. "Gio called" **VS**

Without contextual/discourse cues, sentences like (17) are structurally ambiguous between a null-subject interpretation (17a) and a postverbal subject one (17b), with a marked processing preference for (17a) as compared to (17b) (De Vincenzi, 1991).

Clause Type	**MaxT**	**SumS**
matrix SVO	3/*ha*/*v*'	7
matrix VS	7/*Top*/*Foc*	11
VS unacc	2/*v*P	3
VS unerg	7/*Top*/*Foc*	11

Table 3.3: Summary of MAXT (*value*/*node*) and SUMS by construction, for the trees in Fig. 3.5 and Fig. 3.6. The expected difficulty gradient is SVO $<$ VS, and unacc $<$ unerg.

As summarized in Table 3.3, both MAXT and SUMS predict the correct preferences under Belletti and Leonini (2004)'s analysis, as the Top and Foc heads have to wait for the whole *v*P to be found, before they can be discharged from memory themselves (Fig. 3.5).

3.3.2.3 Unaccusatives vs. Unergatives

Finally, it is interesting to look at declarative sentences containing intransitive verbs of two classes: unaccusatives (18) and unergatives (19).

(18) È arrivato Gio
Is arrived Gio

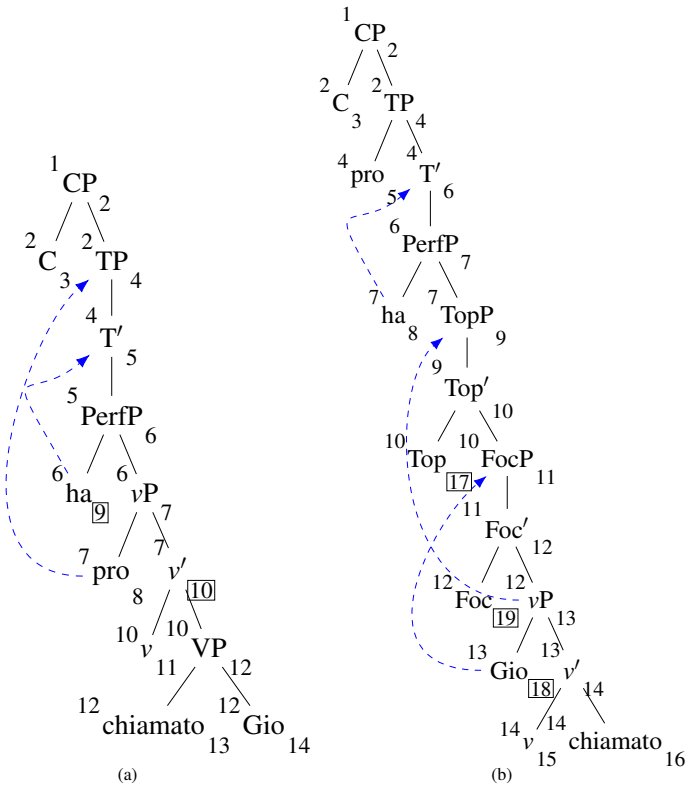


Figure 3.5: Annotated derivation trees for (a) the SVO sentence in (17a), and (b) the VS sentence in (17b).

"Gio arrived" **VS Unaccusative**

(19) Ha corso Gio
Has ran Gio
"Gio ran" **VS Unergative**

While on the surface these sentences look very similar, they differ in that the subject originates in postverbal position for unaccusatives, but in preverbal position for unergatives (Belletti, 1988). Importantly, De Vincenzi (1991) reports faster reading times and higher comprehension accuracy for (18) over (19), a preference that is again correctly captured both by MAXT and SUMS (Fig. 3.6). In particular, due to the fact that unaccusative subjects are base-generated postverbally, MAXT for these constructions is the lowest it can be (2, the tenure of any right sibling which is predicted and immediately discharged; see Table 3.3).

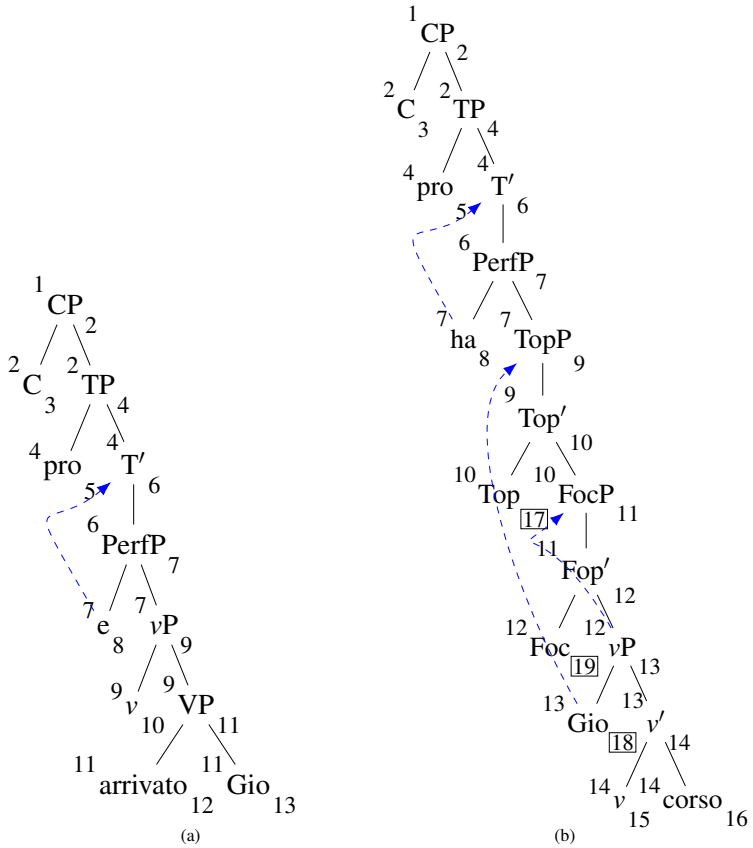


Figure 3.6: Annotated derivation trees for (a) the unaccusative sentence in (18) and (b) the unergative sentence in (19).

Clause Type	<MaxTenure,SumSize>
obj. SRC < ORC	✓
obj. SRC < ORCp	✓
obj. ORC < ORCp	✓
subj. SRC < ORC	✓
subj. SRC < ORCp	✓
subj. ORC < ORCp	✓
matrix SVO < VOS	✓
VS unacc < VS unerg	✓

Table 3.4: Predictions of the MG parser by contrast.

3.4 Discussion

Overall, the success of a top-down parser in modeling the processing difficulties of Italian RCs adds support to the MG model as a valuable theory of how processing cost is tied to structure. A summary of the results in this chapter is shown in Table 4.1.

As already mentioned in Chapter 2, one potential concern with the plausibility of the approach is in the degrees of freedom that are left to the model. In particular, the processing predictions depend on the interaction of three factors: the parsing strategy, the syntactic analysis, and the complexity metrics. Here, I put aside the choice of parsing strategy (but see Hunter, 2018a; Stanojević and Stabler, 2018), and briefly address concerns about the remaining two factors.

Due the large number of existing metrics, it is conceivable that some combination of syntactic analysis and metric could have explained any other processing ranking among sentences. Similarly, it is possible that any syntactic analysis would make the right (i.e., empirically supported) predictions with some metric. Both these possibilities would undermine the relevance of this kind of modeling. Luckily, this does not seem to be the case. In fact, previous work has ruled out the vast majority of the existing metrics, by showing their insufficiency in accounting for some crucial constructions across a variety of possible grammatical analyses (Graf et al., 2017). Thus, it seems that underspecification is not an issue in practice.

The results in this chapter are indeed consistent with these observations, as they show SUMS as a reliable complexity metric. Importantly, as subject-modifying SRCs and ORCs only *tie* on MAXT, these findings are also consistent with Graf et al. (2017)'s hypothesis that SUMS should

be used a secondary metric to adjudicate between constructions, after they tie on MAXT.[4]

A second, reasonable concern is how much the correct predictions depend on the specific syntactic analysis of choice. Due to the richness of existing analyses, here I only considered an analysis of Italian RCs and postverbal constructions which had been extensively referred to in the psycholinguistic literature. To partially address this concern though, I showed how SUMS and MAXT not only make the right predictions for RC constructions under a few different syntactic configurations, but they also correctly account for postverbal subject asymmetries in different kind of sentences. Nonetheless, an important future enterprise will be to look at alternative approaches to postverbal subject configurations, such as *right dislocation* (Antinucci and Cinque, 1977; Cardinaletti, 1998), or *leftward scrambling* (Ordóñez, 1998). Note though that these analyses all assume additional movement dependencies in the structure of ORCps compared to clauses with preverbal subjects. Given what this chapter taught us about SUMS and MAXT, it seems probable that such dependencies would also be picked up by these metrics.

Independently on the specific predictions of the parser for alternative analyses though, the contributions of this line of inquiry would be twofold. From one side, it will improve our understanding of the MG model, by clarifying which aspects of sentence structure drive the parser's performance, and how they weight on the recruitment of memory resources as measured by different metrics. Secondly, grounded in the discriminative power given to MAXT and SUMS by their success across empirical phenomena, comparing the predictions made by the parser for different analyses of the same construction might help adjudicate between competing theoretical assumptions, as was the original goal of Kobele et al. (2013).

Clearly, the fact that the parser relies on an idealized deterministic search strategy is one of the (potentially) most contentious assumption of the MG model, and could thus be used as yet another objection to the plausibility of the linking theory. As already mentioned, the goal is not to claim this as a comprehensive model of processing difficulty, as a cognitively realistic theory would see multiple factors interact with each other to derive the correct contrasts (Demberg and Keller, 2008; Brennan et al., 2016, a.o.). In principle though, the MG parser can be integrated with several of

[4] SUMS by itself does not seem to be enough, as it fails to predict the right preferences for contrasts like English right vs. center embedding (Graf et al., 2017).

these additional factors (e.g., uncertainty; Hunter and Dyer, 2013; Yun et al., 2015). Crucially, the main advantage of the MG model is its transparent specification of the parser's behavior, which clarifies the effects of structural complexity on memory burden and would allow us to separate them from other effects contributing to processing load.

Moreover, while uncertainty is clearly a fundamental component of the human sentence processing system, the fact that an account deliberately abstracting away from all ambiguity can explain effects that would usually be attributed to it is an intriguing result. A fascinating open question is then whether we can characterize those phenomena where ambiguity really is the decisive factor, and cannot be "eliminated" from the model.

Chapter 4

Beyond Processing Asymmetries: Modeling Gradience in Acceptability Judgments

4.1 Introduction

The human judgments linguists use to evaluate the adequacy of syntactic theories fall in a wide, non-binary spectrum of acceptability — a fact well-known from the early days of generative grammar (Chomsky, 1956a, 1965, a.o.). Nonetheless, mainstream syntax has long claimed that grammatical knowledge is, at its core, categorical, and that *gradience* in acceptability judgments comes from extra-grammatical factors (Sprouse, 2007, a.o.). However, the rise of experimental methods in theoretical syntax has renewed the question of whether gradience should be integrated in grammatical theories directly (Keller, 2000; Crocker and Keller, 2005; Sorace and Keller, 2005; Lau et al., 2014, 2015, 2017).

As the relation between grammaticality and acceptability is not transparent, constructing a well-specified theory of how gradient acceptability arises from grammatical knowledge is clearly valuable. From an empirical perspective, however, categorical approaches seem to be at a disadvantage when compared to gradient grammatical models rooted in quantitative, probabilistic frameworks. There is an abundance of well-known proposals about the way syntactic structure and cognitive resources can be integrated to derive connections between acceptability and processing

difficulty (e.g., Yngve, 1960; Wanner and Maratsos, 1978; Rizzi, 1990; Rambow and Joshi, 1994; Gibson, 2000; McElree et al., 2003; Lewis and Vasishth, 2005, a.o.). However, few models based on current grammatical formalisms have been implemented in precise computational frameworks (cf. Boston, 2010).

In order to have a complete theory of how acceptability judgments correlate to categorical grammars, what seems to be necessary is a formal model of the syntactic structures licensed by said grammars, and a theory of how such structures interact with extra-grammatical factors to derive differences in acceptability. This would make it possible to test how assumptions about fine-grained syntactic details lead to quantifiable predictions for the gradient acceptability of individual sentences (Stabler, 2013; Sprouse et al., 2018).

In this chapter, I show how the MG parsing model is effective in addressing these issues. As shown in previous chapters, the MG parser has been successful in studing which aspects of grammar drive processing cost, for a vast set of off-line processing asymmetries cross-linguistically. Moreover, as the results in Chapter 3 demonstrate, the ability of MGs to encode rich syntactic analyses makes the parser especially sensitive to fine-grained grammatical information, and thus able to generate quantitative predictions especially suited to modeling gradience.

Concretely, I model the acceptability judgments for three types of syntactic island effects in English, using as a baseline the judgments reported in (Sprouse et al., 2012a). Importantly, my main aim is not to settle the debate of whether gradience should be found in the grammar itself, or in the interaction between grammar and external factors (if such a debate could ever be settled). What I offer is a formalized, testable model of the latter hypothesis, in the hope of providing ground for a more principled investigation of categorical grammaticality and continuous acceptability.

4.2 Gradience, Acceptability, and Theories of Grammar

Historically, acceptability judgments have played a fundamental role in generative syntax, by providing evidence for the kind of phenomena used to motivate a sound theory of grammatical knowldedge.

> One way to test the *adequacy of a grammar* proposed for [language] L is to determine whether or not the sequences that it generates are actually grammatical, i.e., *acceptable to a native speaker.*
>
> (Chomsky, 1956b)

Interestingly, while the quote above could be interpreted as identifying being *grammatical* with being *acceptable*, generative syntactitian have traditionally used *grammaticality* to refer to the theoretical knowledge of possible structures in the language. However, acceptability judgments are a performance phenomenon, and grammaticality is only one of the many possible factors (e.g. semantic plausibility, processing difficulty, etc.) driving them. Thus, native speakers' judgments about the acceptability of a sentence are clearly only an indirect measure of the categorical distinctions made by the underlying grammar.

Since a speaker's grammar is not directly accessible to observation or measurement, a wide-spread view in linguistics is that acceptability judgments are a reasonable proxy to investigate grammatical knowledge. Because of this non-transparent relation between acceptability and grammaticality though, we need to be careful in how we use human judgments to make inferences about the theoretical properties of the grammar.[1]

In this sense, a fundamental question about the nature of our theories of grammar arises by observing the granularity of acceptability judgments. In particular, modern versions of the Minimalist program presuppose that grammatical knowledge is, at its core, categorical. Under this view, a speaker's grammar establishes a *binary distinction* between the set of well-formed structures, and ill-formed (ungrammatical) ones. However, it is a well-known fact that human do not produce binary acceptability judgments, but instead classify sentences over a continuous spectrum of acceptability.

> An adequate linguistic theory will have to recognize *degrees of grammaticalness* [...]

[1]There is an additional ongoing debate in the literature about the reliability of the acceptability judgments used in theoretical syntax (Linzen and Oseki, 2018; Gibson et al., 2013; Gibson and Fedorenko, 2010; Edelman and Christiansen, 2003). However, this debate is usually concerned with informal data-collection methods focused on the judgment of a single individual. Instead, the data modeled in this chapter were collected during a carefully controlled psycholinguistic experiment. Thus, this debate is at best tangential to the claims of this chapter, independently of the validity of the argument in favor or against informal elicitation techniques (Sprouse and Almeida, 2017; Sprouse et al., 2013; Phillips and Lasnik, 2003; Phillips, 2009).

> there is little doubt that speakers can fairly consistently order new utterances, never previously heard, with respect to their *degree of belongingness to the language*.
>
> (Chomsky, 1975, 131-132)

The question then, is whether categorical grammars can give rise to gradient acceptability judgments. In fact, this apparent discrepancy between presupposed categorical grammaticality and gradient acceptability has led to a variety theories trying to account for it. Here, I want to focus on two main views of how a grammar can account for continuous acceptability, which have been dominating the discussion in the past years.

From one side, one could keep assuming that syntactic competence is categorical, and generates all and only the grammatical structures of a given language. On top of that, there are processing mechanisms of varied nature (e.g., probabilistic prediction, cost of reanalysis, memory load) that give rise to gradience in acceptability. This is the mainstream view in theoretical syntax (Chomsky, 1975; Schütze, 2011, 2016).

On the other hand, one could assume that syntactic knowledge does not establish a binary membership condition over — thus selecting well-formed structures exclusively. Instead, it could generate a probability distribution over all possible — well-formed *and* ill-formed — structures (Crocker and Keller, 2005; Sorace and Keller, 2005; Lau et al., 2017). For instance, Lau et al. (2017, 2015, 2014) argue that the distribution of acceptability across a wide selection of sentences can be predicted by using a variety of probabilistic language models. While probabilistic approaches to grammatical knowledge have been explored even within the generative literature (Keller, 2000; Sorace and Keller, 2005), Lau *et al.*'s take is particularly interesting as it relies on theories of grammar that deviate substantially from modern generative syntax, in the sense that they attempt to explain grammaticality just based on the surface (i.e., word-level) probability of a sentence (see Sprouse et al., 2018, for a critical review of these results).

Importantly, there is no *a priori* reason to prefer one approach over the other. In fact, an idealized version of either of these theories should hypothetically be able to account for the same set of acceptability data.

> Both views have strengths and weaknesses. The probabilistic approach can model certain aspects of linguistic behavior — disambiguation, perception, etc — quite

> easily, but it does not naturally account for intuitions of grammaticality. By contrast, binary categorical models can easily express the distinction between grammatical and ungrammatical sentences, but they construe all sentences in each class as having the same status with respect to well-formedness. They do not, in themselves, allow for distinctions among more or less likely words or constructions, nor do they express different degrees of naturalness.
>
> (Lau et al., 2017, pg. 3)

Because of this, formal models can significantly contribute to the discussion. Importantly, a quantitative approach clarifies what kind of assumptions each of these theories needs to make in terms of complexity of the grammar formalism, and its relations to the mechanisms underlying language acquisition and processing. Thus, well-specified, quantitative implementations of these theoretical frameworks would allow us to understand exactly in which ways they differ.

In this sense though, probabilistic approaches have so far had an empirical advantage, in that they have been directly implemented in quantitative models that could be directly tested over human data. On the contrary, there is a lack of models that take current syntactic assumptions about the nature of the grammar seriously, while also defining a precise performance system that can lead to quantitative predictions.

> By contrast, classical formal grammars cannot, on their own, explain these judgement patterns. In principle, they might be able to do so if they are supplemented with a theory of processing. To date no such combined account has been formulated that can accommodate the data of acceptability judgements to the extent that our best performing language models can.
>
> (Lau et al., 2017, pg. 7)

What is needed to significantly contribute to the debate then, is a model showing how/whether the *right* notion of gradience — fitting human acceptability judgments — can be obtained from a specific categorical grammar.

> [...] there is no single, coherent categorical grammar that is sufficiently formalized to perform the kind of quantitative evaluation that [Lau *et al.*] prefer. [...] direct comparisons between theories are difficult if one (or both) of those theories are insufficiently comprehensive (and formal) to make quantitative predictions

(Sprouse et al., 2018, pg. 595)

This chapter aims to address this gap. Recall that the MG model relates sentence acceptability to sentence structure by specifying: 1) a formalized theory of syntax in the form of MGs; 2) a parser as a model of how the structural representation of a sentence is built from its linear form; 3) a linking theory between structural complexity and acceptability in the form of metrics measuring memory usage. Thus, it seems like this model would be an ideal candidate to test whether a performance system sensitive to fine-grained syntactic analyses can model gradient judgments as measured in sentence acceptability experiments.

4.3 Modeling Gradient Acceptability in Syntactic Islands

The value of a computational modeling approach in addressing theoretical questions is crucially dependent on choosing the right set of test data. When it comes to the MG model discussed in the dissertation, this problem can be reframed as the issue of selecting an appropriate set of contrasts, which are going to be maximally informative when compared through the MG complexity metrics. This was not an issue in Chapter 3, as the contrasts we were interested in modeling were clearly specified by the psycholinguistics literature. In the case of acceptability judgments though, the variety of alternative comparisons forces us to explicitly commit to the nature, and relevance, of the sentences we are going to evaluate (Lau et al., 2017; Sprouse et al., 2018).

In fact, several datasets are available in the literature, created specifically for the study of gradience in acceptability judgments. For instance, we could look at acceptability contrasts present in the datasets used by Lau et al. (2014, 2015, 2017). These datasets collect a range of acceptability judgements over sentences from different domains and languages, obtained by drawing sentences from corpora, automatically inducing grammatical violations, and then crowd-sourcing native speaker acceptability judgements. Alternatively, we could look at more linguistically controlled contrasts, as exemplified in the datasets collected by Sprouse et al. (2013, 2018) or Sprouse and Almeida (2012). All of these contain a vast selection of judgments over sentences with syntactic violations of various sorts, and varying widely in terms of structural configurations.

However, as discussed in previous chapters, the metrics' sensitivity to minor differences in

syntactic structure makes the MG parser's predictions most interpretable when used to compare the relative complexity of minimally different sentences. Careful comparisons across sentences as similar as possible in their underlying syntactic structure seem also desirable, if we want to understand the source of gradient variation in acceptability judgments. For these reasons, this chapter focused on modeling the data on the acceptability of syntactic islands collected by Sprouse et al. (2012a) (henceforth SWP), in a first investigation of the viability of the parser as a model of gradient acceptability.

4.3.1 Gradience in English Island Effects

Syntactic islands are well-known in linguistics (Chomsky, 1965; Ross, 1968) as a set of phenomena in which the acceptability of a sentence is degraded, in relation to the interaction of a long-distance dependency and its syntactic context. Consider the following sentences:

(20) a. What$_i$ do you think that John left t_i at the office?

b. What$_i$ do you laugh if John leaves t_i at the office?

In 20a, there is a long-distance dependency between *what* in sentence initial position, and its lower position as the complement of the verb *left*. In 20b, this same dependency cannot be established, as *what* is inside an adjunct clause (headed by *if*). Thus, 20b is considered ill-formed by native speakers of standard American English. Since establishing long-distance dependencies from inside an adjunct leads to ungrammaticality, adjunct clauses are classic example of island structures.

To investigate the origin of these effects, SWP conducted an extensive study of the acceptability of island constructions, by collecting formal acceptability judgments for four island types using a magnitude estimation task.[2] In particular, the stimuli in SWP's design were based on a (2×2) factorial definition of island effects, and explicitly identify two structural factors that might affect acceptability: 1) the length of a long-distance dependency; 2) the presence of a so-called

[2]Specifically, the task was a standard seven-point scale acceptability-judgment task, with 1 representing the *least acceptable* option, and and 1 representing the *most acceptable* one (Sprouse et al., 2012a). See Marty et al. (2019) and Sprouse and Almeida (2017, a.o) for a discussion of how the choice of task affects the sensitivity/reliability of acceptability judgment experiments.

"island construction" (Kluender and Kutas, 1993). Consider how the sentences in (21) expand on the contrast in (20):

(21) a. Who$_i$ t_i thinks that John left his briefcase at the office? **Matrix | Non Isl.**

b. What$_i$ do you think that John left t_i at the office? **Emb. | Non Isl.**

c. Who$_i$ t_i laughs if John leaves his briefcase at the office? **Matrix | Isl.**

d. What$_i$ do you laugh if John leaves t_i at the office? **Emb. | Isl.**

In (21), the extraction site of the wh-element can either be in the matrix clause (*Matrix*) or in the embedded one (*Emb.*). This contrast is meant to isolate the effect of dependency length (short vs long) on acceptability. Sentences are then modulated along a second dimension: *Island* or *Non Island*. Consider the *Island* sentences in 21c and 21d. Importantly, the Island label does not imply an island violation (i.e., ungrammaticality). Instead, I am following Sprouse *et al.* (who in turn follow Kluender and Kutas (1993)), and refer in these constrasts to the presence of an *island structure*: a clause that could *in principle* be as island (e.g., an adjunct clause, as in 21a,b vs 21c,d). According to Kluender and Kutas (1993), there is a processing cost associated with the operations necessary to build these structures. Decomposing sentences across this second dimension is then meant to isolate the effect of processing *intrinsically costly* island constructions.

Crucially, the careful dimensional decomposition of the test sentences (a *factorial design*) results in minimally different pairwise comparisons, ideal for the MG parser's modeling approach. Moreover, while a categorical grammar would predict a binary split in sentence acceptability (violates an island/doesn't violate an island), the continuous scale of the acceptability estimation task in SWP revealed a spectrum of gradient judgments. Thus, the contrasts in this study are optimal for our purposes.

In what follows, I test whether the gradient of acceptability shown in SWP's data is predicted by a parser grounded in a rich categorical grammar. Before proceeding with the modeling analysis though, it seems to be important to make an additional note about its aim. In particular, given that the data I am going to test the parser on is a collection of acceptability judgments for island effects, it seems important to clarify how this modeling approach fits into an on-going debate about the *nature* of these effects.

4.3.2 Another Debate: The Nature of Island Effects

There is an ongoing debate in the literature about the nature of island effects — with classical syntactic accounts rooting them in grammatical constraints, while others arguing that such effects can be reduced to a conspiracy of processing factors.

Consider again the examples in 21d, which we said is considered unacceptable, since there is a long-distance dependency established from a position within an adjunct clause. Note that this is not an explanation as for *why* that should be the case. Instead, it is an attempt to capture a widely observed generalization: there are specific type of structures that "block" the formation of long-distance dependencies, which have been historically labelled *islands*.

Island phenomena are well attested cross-linguistically, even though there is significant variability in terms of the kind of structures that give rise to these effects (Alexopoulou and Keller, 2003; Saah and Goodluck, 1995; Georgopoulos, 1985; Rizzi, 1980; Andersson et al., 1982, a.o.).

Following Ross (1968), mainstream generative syntax has used these phenomena to posit the existence of island *constraints*: grammaticalized retrictions on the formation of long-distance dependencies. These costraints have played a fundamental role in the development of syntactic theories that assume domain-specific biases in acquisition (Chomsky et al., 1973; Chomsky, 1993, 1986, a.o.). Despite their long history though, island constraints have been lacking stable analyses, in part due to the heterogeneity of these effect and of the acceptability judgments used to motivate different island constraints — often ripe with possible counter-examples (Hofmeister and Sag, 2010; Szabolcsi et al., 2006). Moreover, while many cases of unacceptability involving long-distance dependency formation are collected under the umbrella term of *islands*, it is unclear whether a uniform account of these effects is possible (cf. Sabel, 2002).[3]

These facts have led researchers to formulate a different kind of theories, that attempt to explain island effects not in terms of grammatical constraints, but as side-effects of limitations on the human sentence processing system (Deane, 1991; Hofmeister and Sag, 2010; Kluender, 1993, 1992; Kluender and Kutas, 1993; Alexopoulou and Keller, 2007). That is to say, in 21d there is nothing specifically *wrong* in the formation of a long distance dependency from inside the adjunct

[3]See however Shafiei and Graf (2020) and Graf and De Santo (2019) for recent computational takes attempting to unify these phenomena at a more fundamental level.

clause. Instead, the unacceptability of the sentence would be due to the processing requirements for that specific construction (e.g., increased memory load).

The factorial design in SWP was meant to explicitly address this debate. Specifically, it targeted the idea that the unacceptability of sentences that give rise to island effects could in fact be reduced to the presence of factors that increase processing effort (thus lowering acceptability) independently of specific grammatical constraints. What their experiments showed is that sentences like 21d give rise to a so-called *super-additivity* effect — a non-linear increase in unacceptability, which they argued could not be explained just as the sum of independent processing factors. While SWP take these results as clear evidence agains processing-based accounts of islands, the debate on the source of these effects is still open (cf. Hofmeister et al., 2012a; Sprouse et al., 2012b; Hofmeister et al., 2012b; Sprouse et al., 2016).

Importantly, it looks like the MG model *could* be construed as a processing-based approach to the unacceptability of island phenomena. However, in this chapter I am *not* attempting to reduce island effects to processing demands and, at least at this stage, it is not my purpose to directly engage with this debate. For the same reasons, I do not investigate the *super-additivity* found in SWP's paper. Relatedly, I do not claim that the acceptability of island violations is *purely* syntactic in nature, as it has been shown to be sensitive to a variety of factors (e.g., semantics; Szabolcsi et al., 2006; Truswell, 2011; Kush et al., 2018; Kohrt et al., 2018, a.o.). Crucially, in this chapter I am "just" interested in exploring the idea that the *gradient* component of acceptability judgments arises due to processing factors. The focus on island effects is exclusively due to the optimal baseline offered by SWP's data.

Finally, discussing gradience in the context of island effects could be the source of one additional point of confusion. In the theoretical syntax literature, there is a well-known distinction between *strong* (e.g., Complex NPs) and *weak* (e.g., whether clauses) islands (Ross, 1968; Phillips, 2013a,b, a.o.). Famously, while strong islands never permit extraction, weak islands permit extraction of specific types of elements (Truswell, 2007; Szabolcsi and Lohndal, 2017). For instance, both strong and weak islands disallow extraction of adjuncts, but weak islands are supposed to allow extraction of arguments, under certain conditions. Thus, it looks like islands could be *gradient*, in the sense that there is variation in how severe these effects are depending on the type of extraction

domain, and of the element being extracted. In this sense, it is possible that processing factors could be contributing to such variation (Kluender, 1992, 1998; Keller, 2000), and that a computational model such as the one presented here could help in addressing the origin of such effects (Boston, 2010). These issues are clearly tied to questions about the nature of island constructions. As mentioned repeatedly above though, here I am only interested in addressing the issue of gradience as non-binarity in the degree of acceptability of purportedly grammatical sentences, and *not* in modeling the (un)grammaticality of island violations per se.

Pursuing a deeper understanding of the nature of island effects — and the trade-off between syntactic constraints and memory limitations — is a worthwhile future endeavor, but the relation between categorical grammars and gradient acceptability is a fundamental issue on its own, and it is the focus of the rest of this chapter. Thus, in what follows I exclusively discuss how the MG parser can model the gradience in acceptability for the case studies reported in Sprouse et al. (2012a). I will return to the question of whether this model could give us *any* insights into the question of separating processing and grammatical contributions to island phenomena in Sec. 4.5.

4.4 Modeling Results

SWP focused on English wh-movement dependencies to explore four types of island constructions: Subject, Adjunct, Complex NP, and Whether islands. However, consider the following sentences:

(22) a. What$_i$ do you think that John bought t_i ?

b. *What$_i$ do you wonder whether John bought t_i?

In 22b, *what* originates from within the clause headed by *whether*, thus leading to a whether-island violation and ungrammaticality. Arguably though, there is no difference in *structure* between 22a and 22b, at least at the level of the geometry of the derivation tree. Since the MG parser is only sensitive to structural differences, in what follows I ignore the case of Whether islands and concentrate on the remaining three cases.

As in Chapter 3, here I focus on the predictions made by a ranked version of ⟨MaxT, SumS⟩

Island Type	Sprouse et al. (2012)			Ex. #	MG Parser
Subject Island Case 1	Subj. | Non Isl.	>	Obj. | Non Isl.	23b > 23a	✓
	Subj. | Non Isl.	>	Obj. | Isl.	23b > 23d	✓
	Subj. | Non Isl.	>	Subj. | Isl.	23b > 23c	✓
	Obj. | Non Isl.	>	Obj. | Isl.	23a > 23c	✓
	Obj. | Non Isl.	>	Subj. | Isl.	23a > 23d	✓
	Obj. | Isl.	>	**Subj. | Isl.**	23c > 23d	23c < 23d
Subject Island Case 2	Matrix | Non Isl.	>	Emb. | Non Isl.	24a > 24b	✓
	Matrix | Non Isl.	>	Matrix | Isl.	24a > 24c	✓
	Matrix | Non Isl.	>	Emb. | Isl.	24a > 24d	✓
	Matrix | Isl.	>	Emb. | Isl.	24b > 24d	✓
	Matrix | Isl.	>	Emb. | Non Isl.	24c > 24b	✓
	Emb. | Non Isl.	>	Emb. | Isl.	24c > 24d	✓
Adjunct Island	Matrix | Non Isl.	>	Emb. | Non Isl.	25a > 25b	✓
	Matrix | Non Isl.	>	Matrix | Isl.	25a > 25c	✓
	Matrix | Non Isl.	>	Emb. | Isl.	25a > 25d	✓
	Matrix | Isl.	>	Emb. | Isl.	25b > 25d	✓
	Matrix | Isl.	>	Emb. | Non Isl.	25c > 25b	✓
	Emb. | Non Isl.	>	Emb. | Isl.	25c > 25d	✓
Complex NP Island	Matrix | Non Isl.	>	Emb. | Non Isl.	26a > 26b	✓
	Matrix | Non Isl.	=	Matrix | Isl.	26a = 26c	✓
	Matrix | Non Isl.	>	Emb. | Isl.	26a > 26d	✓
	Matrix | Isl.	>	Emb. | Isl.	26b > 26d	✓
	Matrix | Isl.	>	Emb. | Non Isl.	26c > 26b	✓
	Emb. | Non Isl.	>	Emb. | Isl.	26c > 26d	✓

Table 4.1: Summary of results (as pairwise comparisons) from Sprouse et al. (2012a), and corresponding parser's predictions (x > y: x more acceptable than y).

in comparing memory burden for contrasting sentences.[4] In addition, the core linking hypothesis explicitly connects processing difficulty to acceptability by assuming that higher memory cost implies lower acceptability. Table 4.1 presents a summary of all modeling contrasts in the chapter, compared with the experimental results of SWP.[5]

[4] A summary of how each individual base metric fares on these contrasts can be found in Appendix B.

[5] All scripts are available at https://github.com/CompLab-StonyBrook/mgproc.

4.4.1 Subject Island: Case 1

First, I model Subject islands as in SWP's Experiment 1, comparing 4 sentence types across 2 conditions: subject/object extraction, and island/non-island. Note again that here *Island* does not imply a violation, but refers to the presence of an island structure (Kluender and Kutas, 1993).

(23) a. What$_i$ do you think the speech interrupted t_i? **Obj/Non Island**

b. What$_i$ do you think t_i interrupted the show? **Subj/Non Island**

c. What$_i$ do you think the speech about global warming interrupted the show about t_i? **Obj/Island**

d. What$_i$ do you think the speech about t_i interrupted the show about global warming? **Subj/Island**

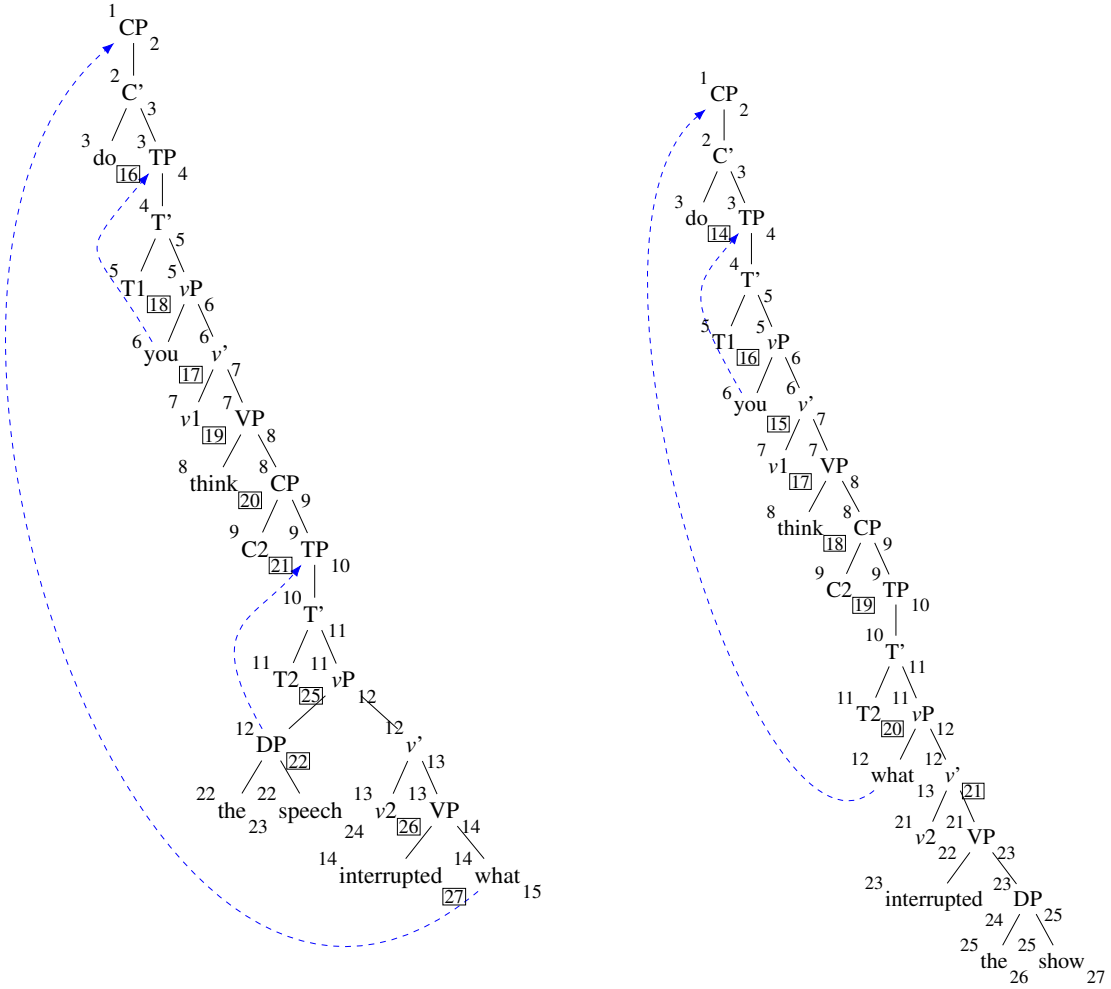
Annotated MG derivation trees for these sentences are shown in Fig. 4.1 (object/subject with no island) and Fig. B.2 (with island).[6] The parser's predictions (via MAXT) overall match the experimental results (see Table 4.1).[7]

The factorial design of the original study helps us understand the model's predictions. The contrast between 23b and 23a, 23d is correctly captured by MAXT. This is due to the wh-element spanning a longer, more complex structure comprising the whole embedded DP subject in the Island cases. Compare 23a and 23b, both with highest tenure on *do* (14 and 11, respectively — Table4.2). In 23a, *do* is conjectured after *what* has been scanned from the input, but it cannot be flushed out of memory until *what* is confirmed in its base position as the embedded complement. In 23b, *do* only has to wait in memory only until the embedded subject position is reached.

Consider now 23c. Here the highest tenure is on the embedded *T* head, which has to wait for the wh-element in object position, and then for the whole complex DP in subject position, before it can finally be flushed out of the queue. The longer wh-dependency in the object case explains once

[6]This chapter provides annotated derivations just for the Subject island case, as an illustrative example. Derivations for all other island types were drawn according to standard minimalist analyses of the test sentences (e.g., Adger, 2003), and can be found in Appendix B. Source files for the MG trees are also available at https://github.com/aniellodesanto/mgproc/tree/master/islands.

[7]When a wh-element is displaced from an embedded position, I avoid intermediate landing sites due to successive cyclicity. As intermediate movement steps do not affect the traversal strategy, this choice does not significantly change the results (cf. Zhang, 2017).



(a) (b)

Figure 4.1: Annotated derivation trees for (a) 23a (object, non-island) and (b) 23b (subject, non-island).

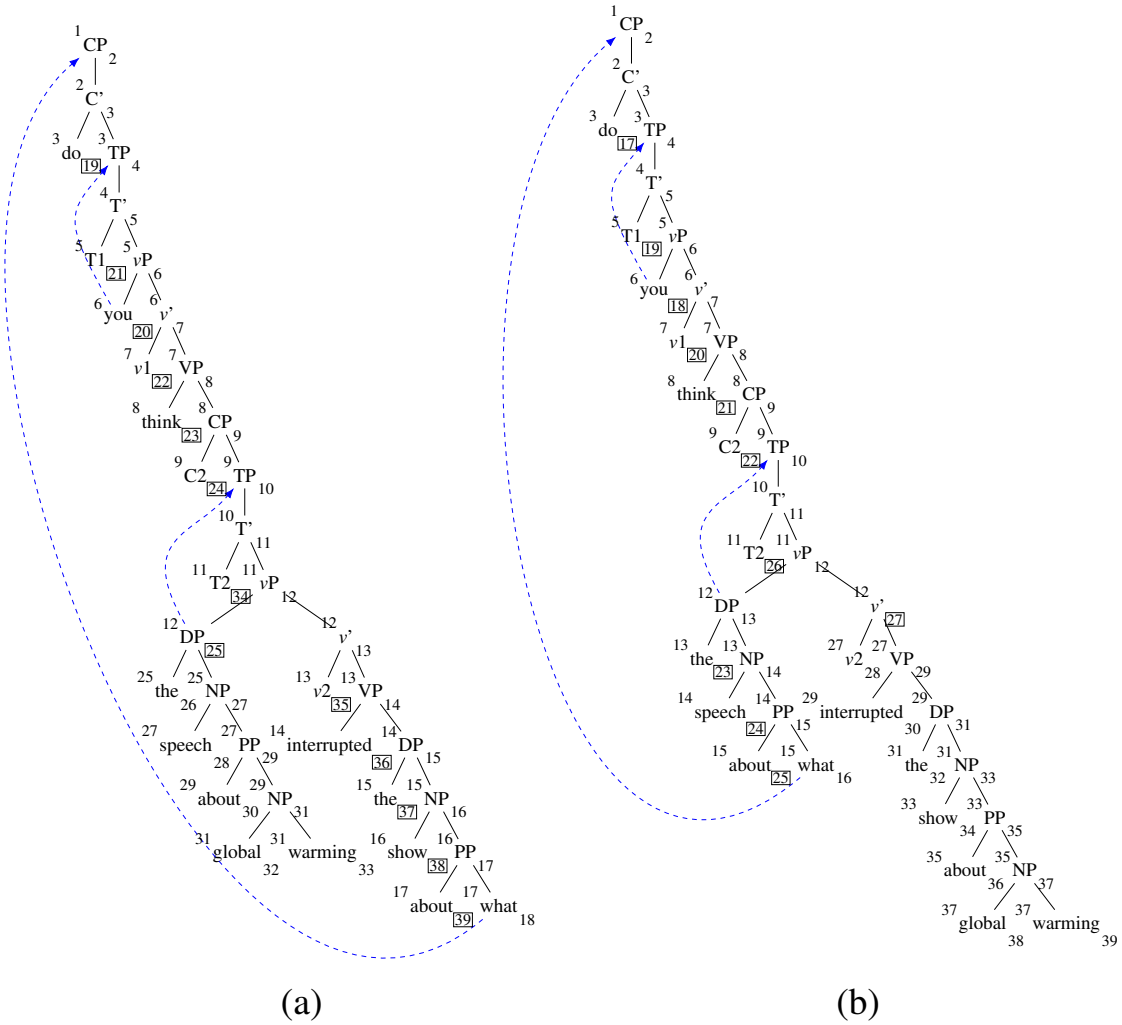


Figure 4.2: Annotated derivation trees for the test sentences in (a) 23c (object, island) and (b) 23d (subject, island).

again why 23b is preferred over 23c, and the additional complexity of the DP subject is crucial in driving the 23a $>$ 23c contrast.

Finally, there is one case in which parser's predictions and experimental data disagree: the contrast between subject and object extraction in the island condition (23c vs 23d). The parser predicts that 23c should be more acceptable than 23d (*Subj/Island* $>$ *Obj/Island*). This is not surprising, as the memory metrics pick up on the additional length of the extraction in the object case, and thus obviously predict the preference for a subject gap. However, SWP show *Obj/Island* $>$ *Subj/Island* — which is expected from a theoretical perspective since 23d is the ungrammatical condition (i.e., there is an extraction out of an island).

I will come back to the significance of this mismatch in Sec. 4.5. For now, recall that here I am not trying to account for the ungrammaticality of island violations. Instead, I am trying to show that the gradience can arise correctly from a processing system fine-tuned to the details of a categorical grammar. Crucially for this claim then, the parser correctly predicts the gradient of acceptability for those conditions that, according to the grammar, should all be equivalent (i.e., those containing no forbidden extraction).

4.4.2 Subject Island: Case 2

The previous section suggests that, when a grammatical violation coincides with processing factors (e.g., length of a dependency), parser and human judgments should match on all contrasts. Luckily, SWP offer us the chance to test such a prediction, with a second set of subject island sentences. SWP's Experiment 2 compares a *short* dependency and *long* dependency (*matrix* vs *embedded* extraction in the original paper), again in an island and non-island condition.

(24) a. Who$_i$ t_i thinks the speech interrupted the primetime TV show? **Matrix | Non Isl.**

b. What$_i$ do you think t_i interrupted the primetime TV show? **Emb. | Non Isl.**

c. Who$_i$ t_i thinks the speech about global warming interrupted the primetime TV show? **Matrix | Isl.**

d. What$_i$ do you think the speech about t_i interrupted the primetime TV show? **Emb. | Isl.**

As expected, parser's preferences and experimental data fully match in this case, as the ungrammatical condition (24d) is also the one in which the movement dependency is the longest. Here however, deriving the correct preferences requires the ranking of ⟨MAXT,SUMS⟩, instead of just MAXT alone (note also that SUMS by itself would not suffice, as it would not predict 24a $>$ 24c, see Table 4.2). Such a ranking also preserves the results in the previous section, which fully relied on MAXT. Interestingly, note how MAXT values for 24b (*Emb.* | *Non Isl.*) and 24c (*Matrix* | *Isl.*) tie here, as the additional structural complexity of 24c does not interact with the main movement dependency (*who* raising from Spec,TP to Spec,CP). Moreover, the *Matrix* | *Non Isl.* (24a) and *Matrix* | *Isl.* (24c) conditions have very similar structures (with an extraction out of the main subject). Nonetheless, the memory metrics are able to capture subtle differences in the way the parser goes through the two sentences (arguably encoding the "island construction" cost of Kluender and Kutas (1993)).

Island Type	Clause Type	Ex. #	MaxT	SumS
Subject Island Case 1	Obj./Non Island	23a	14/*do*	19
	Subj./Non Island	23b	11/*do*	14
	Obj./Island	23c	23/*T2*	22
	Subj./Island	23d	15/*do*	20
Subject Island Case 2	Matrix | Non Isl.	24a	5/*C*	9
	Emb. | Non Isl.	24b	11/*do*	14
	Matrix | Isl.	24c	11/*T2*	9
	Emb. | Isl.	24d	17/*T2*	20

Table 4.2: Summary of MAXT (*value*/*node*) and SUMS by test sentence for Subject island in case 1 and 2 (*T2* marks the embedded *T* head.)

4.4.3 Adjunct and Complex NP Islands

So far, we have been successful in replicating SWP's acceptability judgments via the MG parser. However, we might wonder whether this success is due to something peculiar in the way the Subject island test cases interact with the MG parsing strategy. Thus, I tested the MG parser on Adjunct and Complex NP islands[8], again using as a baseline the results in SWP's Experiment 1.

[8]Technically, "Complex NP" is traditionally used to refer both to an extraction of out the clausal complement of a noun, and out of a relative clause (Boeckx, 2012, a.o.). Here however I use this term to refer exclusively to the former

The test sentences for the adjunct case were as follows:

(25) a. Who$_i$ t_i thinks that John left his briefcase at the office? **Matrix | Non Isl.**

b. What$_i$ do you think that John left t_i at the office? **Emb. | Non Isl.**

c. Who$_i$ t_i laughs if John leaves his briefcase at the office? **Matrix | Isl.**

d. What$_i$ do you laugh if John leaves t_i at the office? **Emb. | Isl.**

As for Subject islands in case 2, $\langle$MaxT,SumS$\rangle$ correctly predicts the pattern of acceptability reported by SWP, matching the empirical results across all conditions (see Table 4.1). Similar results are obtained for the Complex NP island, with test sentences as follows:

(26) a. Who$_i$ t_i claimed that John bought a car? **Matrix | Non Isl.**

b. What$_i$ did you claim that John bought t_i? **Emb. | Non Isl.**

c. Who$_i$ t_i made the claim that John bought a car? **Matrix | Isl.**

d. What$_i$ did you make the claim that John bought t_i? **Emb. | Isl.**

Once more, the parser matches the acceptability preferences reported in SPW correctly in all conditions. Particularly interesting is the absence of a contrast between 25a and 25c. This is again due to the absence of a real interaction between the additional structural complexity of the island and the main movement dependency. The fact that it results in a tie stresses how movement dependencies and structural complexity conspire with the top-down strategy of the MG parser in non-trivial ways to drive memory cost.

4.5 Discussion

This chapter argued for an MG parser as a good, non probabilistic formal model of how gradient acceptability can be derived from categorical grammars. In doing so, it provides one of the first quantitative models of how processing factors and fine-grained, minimalist-like grammatical information can conspire to modulate acceptability. As a proof-of-concept, I replicated the gradient

case, consistently with SWP's notation.

Island Type	Clause Type	Ex. #	MaxT	SumS
Adjunct Island	Matrix | Non Isl.	25a	13/*PP*	10
	Emb. | Non Isl.	25b	17/*PP*	18
	Matrix | Isl.	25c	13/*PP*	11
	Emb. | Isl.	25d	21/*PP*	28
Complex NP Island	Matrix | Non Isl.	26a	5/*C*	9
	Emb. | Non Isl.	26b	13/*did*	19
	Matrix | Isl.	26c	5/*C*	9
	Emb. | Isl.	26d	15/*did*	21

Table 4.3: Adjunct Island and Complex NP Island: MAXT (*value*/*node*) and SUMS values by test sentence.

acceptability scores for the island effects in Sprouse et al. (2012a). The success of the parser on this baseline is encouraging, and opens new research questions in several directions.

Obviously, the target judgments modeled here are part of a restricted set. Future studies in this sense will benefit from wider comparisons among minimally different variants of acceptable and unacceptable sentences (Sprouse et al., 2013, 2016). As mentioned, the nature of the model makes comparisons beyond pairs of minimal sentences hard to interpret. However, in future it might be possible to define normalization measures for memory metrics computed over sentences with widely different underlying structures.

In Section 4.3 I avoided discussing the nature of island effects, as I did not mean for the MG model to directly address the debate of whether island violations are reducible to processing factors, or are instead tied to core grammatical constraints. Importantly, while this approach might superficially be construed as a reductionist theory, it is not: for instance, the MG parser by itself is not able to explain the difference between sentences that are simply hard to process, and sentences considered unacceptable/ungrammatical. Thus, the model is theoretically neutral with respect to grammatical or reductionist frameworks.

However, consider the first case of Subject islands we analyzed in Sec. 4.4. The parser produced the right predictions for all test sentences except when, in the presence of an island construction, the longest movement dependency and the island violation did not coincide (23c and 23d). This mismatch is not only explained, but it is actually expected, if we embrace a grammatical theory of island constraints. Under such theory, 23d is preferable from a processing perspective (as

it involves shorter dependencies), but its acceptability is lowered by the fact that it violates a grammatical constraint, while 23c does not. While we have to be careful in formulating hypotheses based on a single data point, this contrast suggests that the MG model could help us investigate those aspects of acceptability that are fundamentally tied to grammatical constraints. Moreover, recent computational work offers ways to incorporate syntactic constraints into an MG parser directly, and explore the relation between grammatical constraints and the complexity of the parse space (Graf and De Santo, 2019).

As mentioned in Section 4.2, many hypotheses have been formulated in the past about the way memory and grammatical factors conspire to produce processing differences across sentences. Thus, it is reasonable to wonder what are the benefits of the particular linking hypothesis implemented here. As pointed out before, one of the main advantages of our model is the tight connection between the parser behavior and the rich grammatical information encoded in the MG derivation trees. This allows for rigorous evaluations of the cognitive claims made by modern syntactic theories. In the future, it would be interesting to see whether SPW's results can be derived from different cognitive hypotheses; for instance by implementing in the MG model the variety of constraints explored by Boston (2012) for a dependency parser. Moreover, in this dissertation I employ a deterministic parser to exclusively focus on the relation between structural complexity and memory usage. However, it is known that structural and lexical frequency influence islands' acceptability (Chaves and Dery, 2019, a.o.). Thus, informative insights would come from implementing information-theoretical complexity metrics over the MG parser (Hale, 2016), and explore the predictions of expectation-based approaches.

Finally, another advantage of having a computational model which provides a testable link between syntactic theory and behavioral data, is that it allows us to integrate structural hypotheses in existing psycholinguistic theories in a way that leads to precise quantitative predictions.

In line with recent work using the MG parser as a model of processing difficulty, Section 4.4 focused on the predictions made by MaxT and SumS. Clearly, one could easily conceive of metrics that take different syntactic information into account (for example, by counting the amount of bounding nodes or phases). However, tenure and size arguably rely on the simplest possible connection between memory, structure, and parsing behavior — as they exclusively refer to the

geometry of a derivation tree, without additional assumptions about the nature of its nodes.

Of course, a question remains about the cognitive plausibility of such metrics. While this model is certainly not the first to formalize memory cost as associated to the length of movement dependencies, the previous discussion highlighted how size-centered metrics do not simply depend on the length of a movement steps. Instead, they pick up on the non-trivial changes in the behavior of the parser, based on how long-distance dependencies interact with local structural configurations. Thus, they cannot trivially be identified with other length-based measures (cf. Gibson, 1998; Rambow and Joshi, 1994, a.o.). However, the complexity metrics exploited by the MG parser generally rely on very weak assumptions about the nature of human memory. In a sense, this could be considered a perk, as it leaves the model open to connections with a variety of sentence processing theories. In another sense though, this lack of cognitive plausibility weakens the impact of the approach, as it is often difficult to connect its results to more general concerns in the sentence processing literature. Chapter 5 will discuss various ways to re-evaluate the existing complexity metrics in light of psychological insights about human memory mechanisms.

Chapter 5

Extending the Model: The Case of Syntactic Priming

5.1 Introduction

At this point in the dissertation, it should be clear how the MG parser and its complexity metrics can be used to account for processing asymmetries.

While the MG model is able to account for a variety of phenomena cross-linguistically, Chapter 2 mentioned a set of processing effects the model in its current implementation seems unable to capture. Specifically, Zhang (2017) argues that the existing MG metrics are unable to reproduce the complexity profiles she found for the processing of *stacked relative clauses* (RCs) in English and Mandarin Chinese.

Consider the following example of a stacked RC construction, in which a noun phrase (*the reporter*) is modified by two relative clauses — an ORC (RC_1), and an SRC (RC_2):

(27) The reporter [$_{RC_1}$ who the senator attacked *t* last year] [$_{RC_2}$ who *t* received the Pulitzer yesterday] is now facing a public trial.

In such cases, the parser will have to resolve a dependency between *the reporter*, and two integration sites within the RCs: one in object position (RC_1), and one in subject position (RC_2).

Interested in the processing profile for this kind of constructions, Zhang conducted a series of

self-paced reading experiments showing that stacked relatives are processed faster when RC_1 and RC_2 are of the same type (e.g., both ORCs, as in 28) than when they are of different types (as in 27).

(28) The reporter [$_{RC_1}$ who the senator attacked *t* last year] [$_{RC_2}$ who the actor pushed *t* yesterday] is now facing a public trial.

The intuition here seems to be that having seen the first RC induces *facilitatory effects* in processing the second RC, when these have the same underlying syntactic structure.

According to Zhang, no existing metric can account for this *parallelism* effect. This is not surprising *per se*, as the MG parser in its current implementation cannot keep track of the relation between two structurally independent clauses. However, this result is made even more compelling by the fact that the MG parser has been strikingly successful in accounting for the preference of simple SRC over ORC cross-linguistically. Crucially, the processing profiles of stacked RCs seem to reflect a wider, well-observed psycholinguistic phenomenon known as *syntactic priming*.

Syntactic priming (also *structural priming* or *structural persistence*; Tooley and Traxler, 2010) refers to cases in which processing of a *target* sentence is facilitated following processing of a *prime* sentence with the same syntactic structure. For instance, an ORC preceded by another ORC is easier to process than an ORC preceded by a SRC. Thus, we would expect 29b to be easier than 30b:

(29) a. The horse that the lions chased **ORC prime**

b. The mouse that the chicken kissed **ORC target**

(30) a. The horse that chased the lions **SRC prime**

b. The mouse that the chicken kissed **ORC target**

Facilitatory effects linked to structural repetition have been extensively studied in production, and have also been found in comprehension (Tooley and Traxler, 2010; Thothathiri and Snedeker, 2008; Luka and Barsalou, 2005). While general priming phenomena are well-known and well-attested though, results on the specific timing and triggers of such effects are still controversial, and the exact mechanisms responsible for them are topic of extensive debate.

Moreover, as for Zhang (2017)'s stacked RCs, it is doubtful that the MG model would be able to account for these effects. In the current implementation, no MG metric can keep track of the parser having encountered an identical structure twice. On the contrary, at least intuitively, having two complex structures (e.g., two ORCs) should weigh *more* for an MG derivations than having a simple structure followed by a more complex one (e.g., an SRC followed by an ORC), thus making the opposite processing prediction. In order to capture these interactions across similar structures, the model would need to consider how successive occurrences of identical movement types might affect their overall memory cost.

Following insights from the psycholinguistic literature on priming, in this chapter I propose extensions to the MG model, that should be able to account for *structural repetition* effects. In order to do so, it will be necessary to re-think the way memory burden is measured by the MG parser.

The rest of the chapter is structured as follows. First, I show how existing MG metrics are not able to provide a satisfactory account for the processing contrasts reported for stacked RCs and for a selected set of priming effects. Then, in order to implement a notion of *memory reactivation*, I introduce a new set of metrics sensitive to repetitions of movement features. I discuss how these new metrics derive the correct predictions when we consider the new processing effects in isolations. I then show how they fail if we also consider a variety of baseline phenomena previously accounted for by the MG model. However, being able to predict processing profiles across multiple phenomena is fundamental in order to maintain good empirical coverage for the model. Thus, in the final part of the chapter, I suggest an approach to measuring memory load that discards the current strict-ranking method of metric evaluation, in favor of an unranked, weighted system.

5.2 Limits of the MG Model: Test Cases

In the rest of the chapter, I will test the model's predictions for stacked RCs as reported in Zhang (2017), and evaluate priming effects for English subject and object relative clauses (Brandt et al., 2017; Hutton and Kidd, 2011). Importantly, in exploring the limits of the MG parser as an effective model of human sentence processing, we want to look at phenomena that can offer new insights

into why a specific metric succeeds or fails over a certain construction.

In this sense, stacked RC and primed RC effects seem to be an ideal litmus test for the performance of the MG model in its current implementation. From one hand, these constructions involve asymmetries in the processing of RCs, a phenomenon that is well understood from an MG parsing perspective. On the other hand, the facilitatory phenomena linked to structural parallelism seem to be due to memory mechanisms — like the tracking of identical syntactic constructions — outside of the reach of the current MG processing implementation. The aim of this section is to introduce the exact set of test cases that will be the target of the modeling simulations in the rest of the chapter.

5.2.1 Stacked RCs

As mentioned above, stacked RCs are constructions in which a relativized noun phrase is modified by two relative clauses. Zhang (2017) explored the processing of such constructions in English and Mandarin Chinese, in a 2×2 design crossing extraction type (subject or object) with the position of the RC (RC_1 or RC_2). Considering both languages can give us important insights into the functioning of the model, due to the different position occupied by a RC with respect to the noun: postnominal (English), and prenominal (Mandarin). To match Zhang (2017)'s experimental data, the parser should predict for both languages faster reading times when RC_1 and RC_2 are of the same type, than when they are of different types: $SS < OS$ and $OO < SO$.

In my simulations, I consider test cases for English as in (31), and Mandarin Chinese as in (32):

(31) a. The horse that kicked the wolf on Tuesday that patted the lion just now went home **SS**

b. The horse that the wolf kicked on Tuesday that patted the lion just now went home **OS**

c. The horse that kicked the wolf on Tuesday that the lion patted just now went home **SO**

d. The horse that the wolf kicked on Tuesday that the lion patted just now went home **OO**

(32) a. Nage zai xingqier tile xiaoma haojici de zai jintian zhuile
DEM on Tuesday kick-PERF horse several-times REL on today chase-PERF
daxiang de gongniu likaile jia
elephant REL bull leave-PERF home

'The bull that kicked the horse for several times on Tuesday that chased the elephant earlier today left home.' **SS**

b. Nage zai xingqier xiaoma tile haojici de zai jintian zhuile
DEM on Tuesday horse kick-PERF several-times REL on today chase-PERF
daxiang de gongniu likaile jia
elephant REL bull leave-PERF home
'The bull that the horse kicked for several times on Tuesday that chased the elephant earlier today left home.' **OS**

c. Nage zai xingqier tile xiaoma haojci de zai jintian daxiang
DEM on Tuesday kick-PERF horse several-times REL on today elephant
zhuile de gongniu likaile jia
chase-PERF REL bull leave-PERF home
'The bull that kicked the horse for several times on Tuesday that the elephant chased earlier today left home.' **SO**

d. Nage zai xingqier xiaoma tile haojci de zai jintian daxiang
DEM on Tuesday horse kick-PERF everal-times REL on today elephant
zhuile de gongniu likaile jia
chase-PERF REL bull leave-PERF home
'The bull that the horse kicked for several times on Tuesday that the elephant chased earlier today left home.' **OO**

For reference, the first letter in each acronym indicates the type of the first RC, and the second letter the type of the second RC — for instance, SS stands for an *SRC* stacked above an *SRC*.

5.2.2 Priming Subject and Object RCs

As mentioned before, in choosing test cases for *syntactic priming* phenomena, it is important to focus on effects that are clearly ascribable to structural overlaps between the prime and the target.

Some of the best known priming studies investigate cases such as prepositional-dative vs double-object alternations, active-passive alternations, or garden-path sentences (Pickering and Ferreira, 2008). However, it is unclear whether such effects are due to the repetition of internal structural configurations, as these constructions either involve argument alternations that introduce confounds between surface syntax and thematic mappings (Ziegler et al., 2017; Ziegler and

Snedeker, 2018; Oltra-Massuet et al., 2017), or might even be reducible to lexical priming (Traxler et al., 2014).[1]

Thus, I conduct a first set of simulations on a priming phenomenon I believe can be most informative to the model. As mentioned, the MG parser has already been incredibly successful in accounting for the general preference for subject relatives over object relatives (Graf et al., 2017, a.o.). Moreover, SRC vs. ORC preferences have been linked to well-established priming effects strongly indicative of syntactic priming (cf. Troyer et al., 2011; Reali and Christiansen, 2007). For this reason, these constructions provide a solid base on which to evaluate limits and possible extensions of the MG model.

Consider the following template sentences for the SRC and the ORC:

(33) a. The reporter $[_{RC}$who t attacked the senator] admitted the error. **SRC**

b. The photographer $[_{RC}$ who the actor attacked t] admitted the error. **ORC**

According to syntactic priming principles, there is a facilitatory effect on primed targets, so that an ORC preceded by an ORC (*hence*, primed) is easier than an ORC preceded by an SRC (e.g.,34 $<$ 33; Brandt et al., 2017; Hutton and Kidd, 2011).

(34) a. The reporter $[_{RC}$who the senator attacked t] admitted the error. **ORC**

b. The photographer $[_{RC}$ who t attacked the actor] admitted the error. **ORC**

Note that some literature also reports that a subject advantage persists also in primed cases (so that a primed SRC should be better than a primed ORC), while others report that this advantage disappears with priming. Moreover, most experimental studies avoid comparing unprimed cases (e.g., an SRC preceding an ORC vs. an ORC preceding an SRC). In principle, it is unclear whether in these cases we should expect a tie — as both targets are unprimed — or whether we should still see an SRC advantage. Since the empirical evidence for these contrast is contradictory or missing, I will avoid this comparison in the upcoming simulations.

[1] In fact, I will generally ignore examples of priming phenomena that could be associated to the repetition of identical lexical items (the so-called *lexical boost*; Traxler et al., 2014)

5.3 Modeling Choices

As discussed in previous chapters, the MG model's sensitivity to fine-grained structural differences makes choosing a syntactic analysis a fundamental component of the approach. Since this chapter is focused on probing the empirical limits of the model and proposing possible alternatives, it seems advantageous to consider multiple accounts of the core constructions under consideration.

As the novel test cases involve subject and object relative clauses, in this section I discuss the details of the syntactic analyses chosen to evaluate these constructions. I discuss two analysis for single and stacked RCs — in English and in Mandarin Chinese.

The single RC constructions are relevant for the priming cases, but also for a series of previously studied processing contrasts, which I will use as a baseline against the new test cases. Similarly, looking both at English and Mandarin will allow us to explore how these analyses differ for languages that have postnomial RCs (English, Italian), and languages in which RCs are prenominal (Mandarin, Japanese, Korean).

Finally, I discuss how to use coordinate structures to build derivations for the priming cases that can be fed to the MG parser.

5.3.1 Choosing a Syntactic Analysis of RCs

Consistently with Graf et al. (2017) and Zhang (2017), I consider two analyses of RC constructions: a promotion analysis (Kayne, 1994), and a wh-movement analysis (Chomsky, 1977). In what follows, I briefly summarize the core assumptions behind these analyses, both for single RC, and for stacked RCs.

5.3.1.1 Promotion Analysis

The details of Kayne (1994)'s promotion analysis for post-nominal languages like English (or Italian) were already discussed in Chapter 3. This analysis presupposes that the RC is the complement of a determiner head. Consider the English RC:

(35) 'The horse that the wolf chased.'

For the sentence in (35) the derivation proceeds as follows: The head noun starts out as an argument of the embedded verb, and undergoes movement into the specifier of CP; the whole CP is then merged with D (see Fig. 5.1).

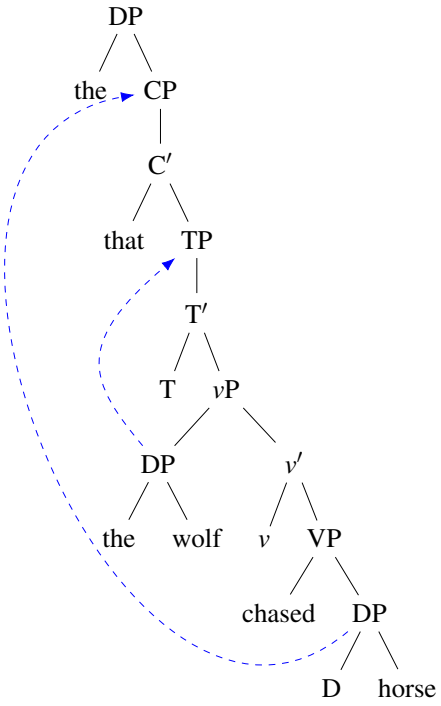


Figure 5.1: Kayne's promotion analysis for relative clauses in postnominal languages as in Fig. 5.1.

Consider now the Mandarin RC in (36):

(36) [$_{RC}$ dahuilang zhui-le de]
wolf chase-PERF REL horse
'The horse that the wolf chased.'

Differently than English, in Mandarin Chinese the RC *dahuilang zhuile de* is prenominal: it precedes the relativized NP *xiaoma*. The RC boundary is marked by the relativizer *de*: this is not equivalent to relative pronouns like *that* or *who* in English, as *de* cannot be used as a pronoun in any other syntactic context.

The derivation of a Chinese RCs under a promotion analysis proceeds as in Figure 5.2. As in English, the relativized NP moves from its base position to Spec,CP. However, in order to get the

correct word order, there is then remnant movement of the whole TP to a higher position: the specifier of the RelP headed by *de*. This analysis can easily be generalized to other prenominal languages like Japanese and Korean.

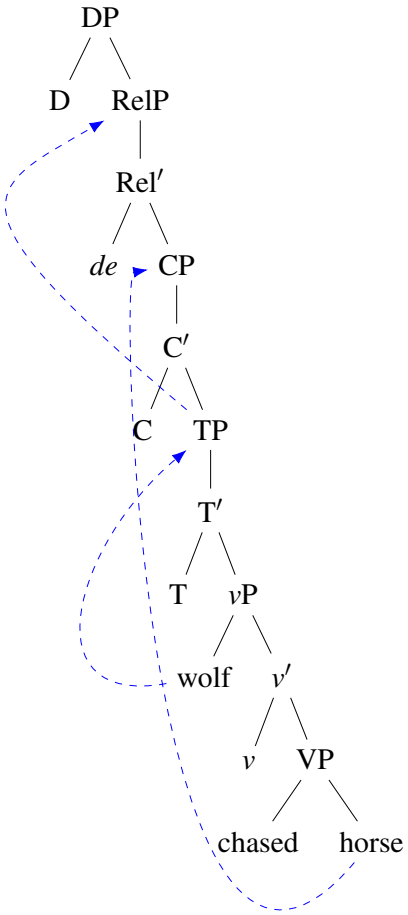


Figure 5.2: Kayne's promotion analysis for Mandarin relative clauses as in (36). In Mandarin Chinese, *de* is an overt relativizer.

I then follow Zhang (2017) in adapting the promotion analysis to English (as in 37), and Mandarin Chinese (as in 38) stacked RCs.

(37) 'The horse [$_{RC1}$ that the wolf chased] [$_{RC2}$that the elephant kicked].'

(38) [$_{RC1}$ dahuilang zhuile de] [$_{RC1}$ daxiang tile de] xiaoma
wolf chase-PERF REL elephant kick-PERF REL horse
'The horse [$_{RC1}$ that the wolf chased] [$_{RC2}$ that the elephant kicked].'

A derivation for the sentence in (37) proceeds as follows: the RC linearly closer to the relativized NP (RC_1, *that the wolf chased*) is generated as the lower RC, containing the relativized NP (*horse*). Then, the relativized NP raises to Spec,CP of the higher RC (RC_2, *that the elephant kicked*), carrying RC_1 over (see Figure 5.3).

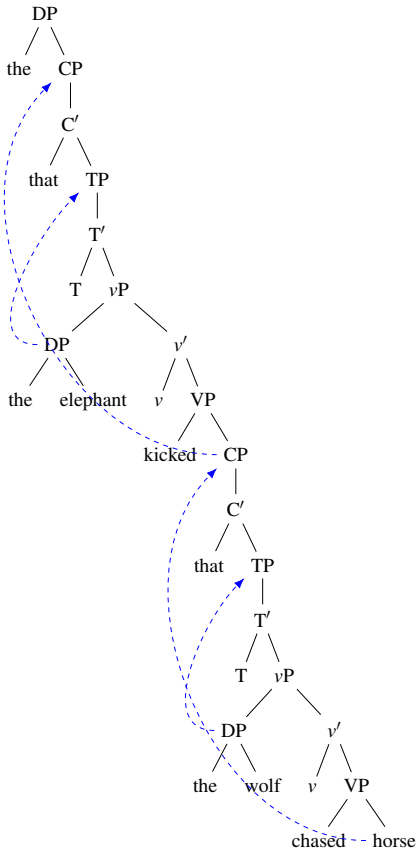


Figure 5.3: Kayne's promotion analysis for stacked RCs in postnominal languages.

In prenominal languages, the derivation is complicated by the sequence of remnant movement operations necessary to arrive at the correct word order (Figure 5.4). Here, the relativized NP first moves from its base position to the Spec,CP position inside the *lower* relative clause (RC_2, in this case). Then, there is remnant movement of the lower TP (*the elephant kicked*) to the specifier of the lower RelP. This RelP is selected by the verb (*chased*) of the higher relative clause, and then moves to the higher Spec,CP. Finally, the remnant of the higher TP (*the wolf chased*) moves to the specifier of the higher RelP.

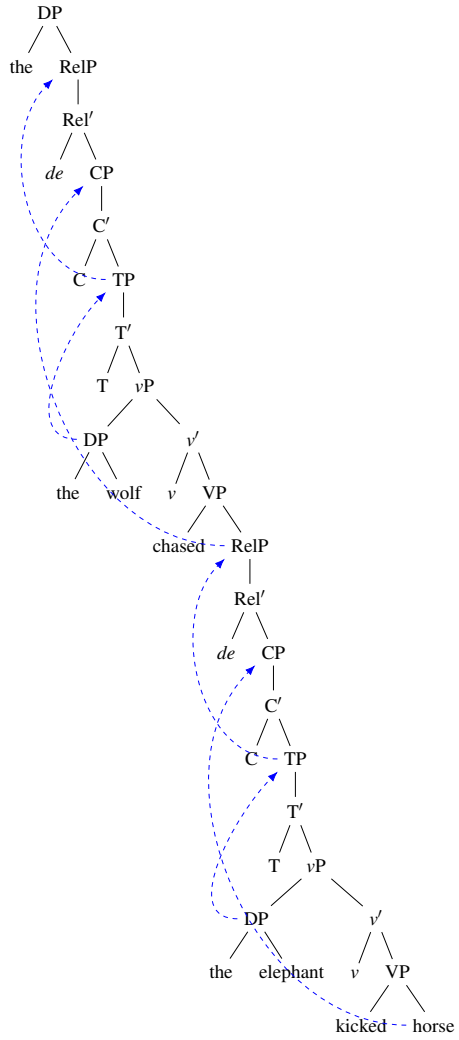


Figure 5.4: Kayne's promotion analysis for stacked RCs in prenominal languages.

5.3.1.2 Wh-movement analysis

The second approach to the structure of RCs that I will consider during the modeling simulations is Chomsky (1977)'s *wh-movement analysis*. As before, in what follows I briefly discuss the details of this analysis for single and stacked RCs in English and Mandarin Chinese.

As the name says, Chomsky (1977)'s approach takes RCs as an instance of wh-movement. Under this view, the RC is treated as an adjunct to the relativized NP. As shown in Figure 5.5a, a silent wh-operator (*Op*) is base-generated within the RC, and then moved from its base position to the specifier of CP. The whole CP, headed by *that*, then merges with the relativized NP as an

adjunct.

Again, things are slightly different for Mandarin. First of all, Mandarin is a *wh-in-situ* language: wh-words do not move from their base-position for question formation. Moreover, the relative marker *de* is not a wh-word.

Thus, the wh-analysis is reduced to the RC adjoining to the relativized NP, and then moving higher across the noun in order to produce the correct word order. Crucially, *de* is here base-generated inside the RC, and then moves to the right of the CP (see Figure 5.5b).

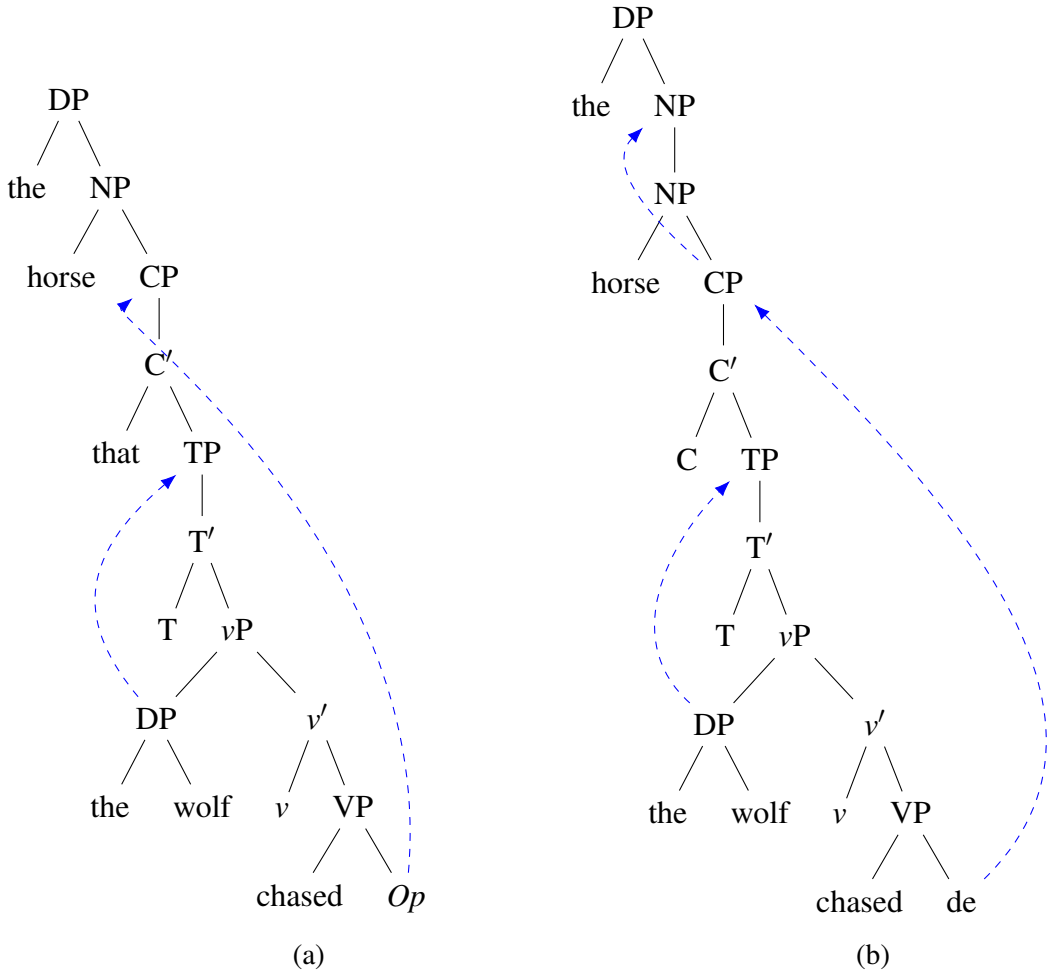


Figure 5.5: Wh-movement analysis for (a) single RCs in postnominal languages, and (b) Wh-movement analysis for single RCs in prenominal languages..

5.3.2 Coordinating RC Targets and Primes

As a reminder, the psycholinguistics results to match for the priming cases I am interested in are:

- $SS < OS$
- $OO < SO$

where, following the convention I used above for stacked RCs, the first letter in each acronym stands for the prime, and the second letter for the target — hence, SS stands for *SRC* priming an *SRC*, OS stands for *ORC* priming an *SRC*, and so on.

From a methodological perspective, it is important to note that in psycholinguistic experiments showing priming effects on RCs, the prime and the target are usually presented as separate sentences (cf., Potter and Lombardi, 1998). Facilitatory effects are then measured on the target alone.

As the parser is not equipped to maintain memory steps across different parses though, prime and target must be part of the same tree (i.e., as coordinated clauses). This does not seem to be too big of a stipulation, given that facilitatory effects due to identity in structural configurations have been reported across conjuncts and embedded clauses as well (Sturt et al., 2010; Callahan et al., 2010; Potter and Lombardi, 1998, i.a.).

However, embedding prime and target in a coordinate structure leads to at least two possible structural configurations:

Case 1 [The reporter [$_{RC1}$ who ....] and the photographer [$_{RC2}$ who ...]] admitted the error.

Case 2 [The reporter [$_{RC1}$ who ....] admitted the error] and [the photographer [$_{RC2}$*who*...]] admitted the error].

In general, it would be reasonable to assume that these differences in the coordinate structure might lead to significant different in metric values. Consider Figure 5.6 and Figure 5.7, showing the structures for these two cases — using ORCs both for the prime and the target, under a promotion analysis of RCs. As one can see from the derivations, the main difference is that in examples following the Case 1 approach, the ConjP containing both the prime and the target needs to raise

to Spec,TP of the main clause. However, there is no significant difference in the structure of the RCs *per se*. Thus, it turns out that this choice doesn't affect the performance of the model.

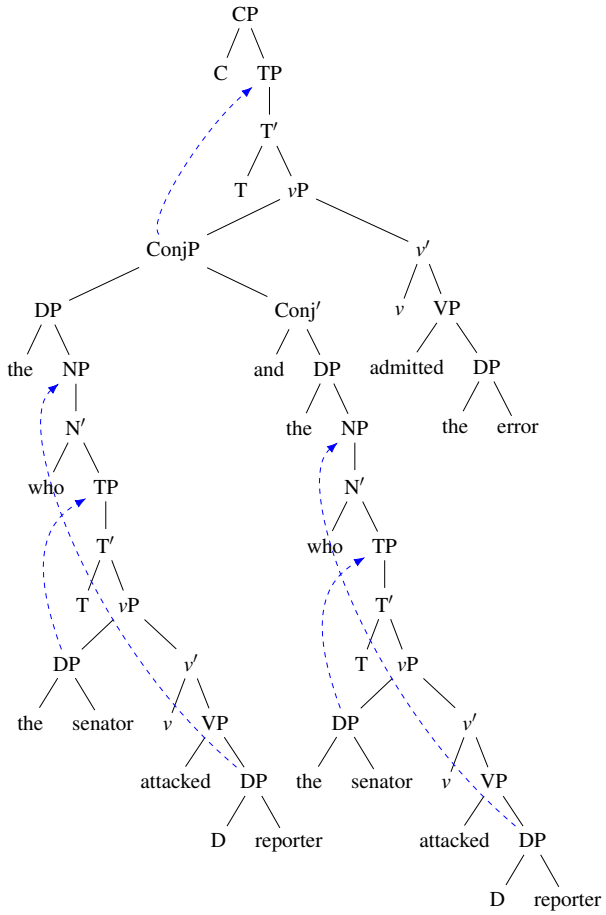


Figure 5.6: Two ORCs in a coordinate structure as in Case 1. RCs are built following the promotion analysis.

In order to keep this chapter easy to follow, in what follows I will only discuss the specific performance of the metric on constructions as in Case 2. Note however that all the same simulations were also performed for Case 1.

In practice, the following test cases were used for the priming simulations:

(39) **SS**

a. The reporter [$_{RC_1}$ who *t* attacked the senator] admitted the error **SRC prime**

b. and the photographer [$_{RC_1}$ who *t* attacked the actor] admitted the error **SRC target**

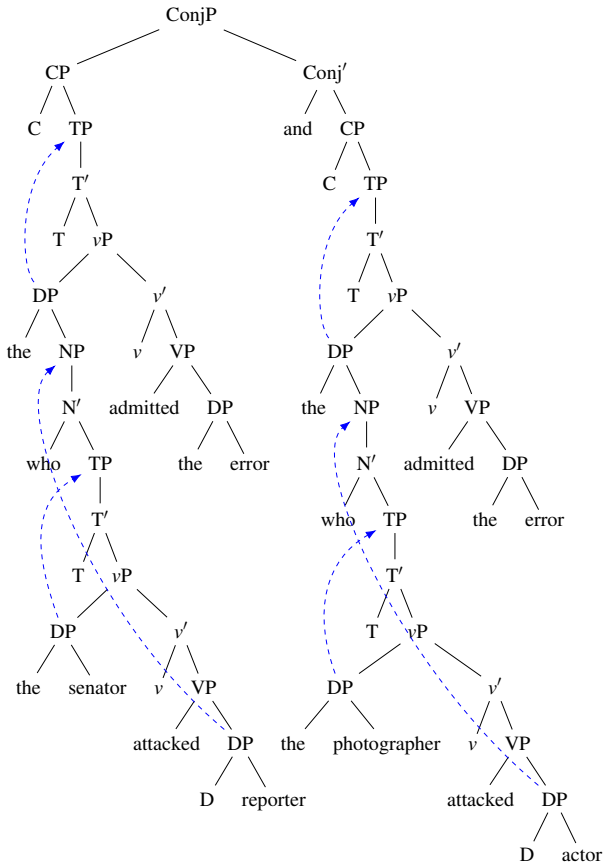


Figure 5.7: Two ORCs in a coordinate structure as in Case 2. RCs are built following the promotion analysis.

(40) **OS**

a. The senator $[_{RC_1}$ who the reporter attacked *t*] admitted the error **ORC prime**

b. and the photographer $[_{RC_1}$ who *t* attacked the actor] admitted the error **SRC target**

(41) **SO**

a. The reporter $[_{RC_1}$ who *t* attacked the senator] admitted the error **SRC prime**

b. and the actor $[_{RC_1}$ who the photographer attacked *t*] admitted the error **ORC target**

(42) **OO**

a. The senator $[_{RC_1}$ who the reporter attacked *t*] admitted the error **ORC prime**

b. and the actor $[_{RC_1}$ who the photographer attacked *t*] admitted the error **ORC target**

5.3.3 Summary of Target Contrasts

Table 5.1 summarizes the new set of processing preferences introduced so far, with the corresponding example sentences that will be used in the MG modeling.

Effect Type	Language	Processing Contrast	Example #
Primed RCs	English	$SS < OS$	$39 < 40$
		$OO < SO$	$42 < 41$
Stacked RCs	English	$SS < OS$	31.a < 31.b
		$OO < SO$	31.d < 31.c
	Mandarin	$SS < OS$	32.a < 32.b
		$OO < SO$	32.d < 32.c

Table 5.1: Summary of processing preferences for the priming and stacked RCs effects modeled in this chapter.

Furthermore, recall we are interested in evaluating the MG approach as a general model of sentence processing. To confirm that the model retains strong empirical coverage, we want to find a set of metrics that accounts for the new phenomena under investigation, while still successfully modeling processing effects predicted by the MG parser in the past.

Procedurally, we need to make sure that there exist a set of metrics that can account for multiple types of processing phenomena cross-linguistically. In what follows, I will consider the following set of phenomena as a *baseline* for new processing effects:

- English Right Embedding < Center Embedding (Kobele et al., 2013; Gerth, 2015)
- SC/RC < RC/SC

  A sentential complement containing a relative clause is easier to process than a relative clause containing a sentential complement (Graf et al., 2015b; Graf and Marcinek, 2014; Gerth, 2015)
- SRC < ORC (Graf et al., 2017)
  - English
  - Korean
  - Japanese

- ORC $<$ SRC (Zhang, 2017)
  - Mandarin Chinese

The reader is referred to Chapter 2 for a recap of how the model fares on these effects, and to Graf et al. (2017) and Zhang (2017) for a detailed discussion.[2] For each of the above phenomena involving a RC construction, I will evaluate all contrasts both under a promotion analysis, and a wh-analysis of RCs.

5.4 Current MG Implementation: Model Evaluation

With all preliminaries in place, the first thing to do is confirm that the current version of the MG model is indeed unable to account for priming effects. In order to present a comprehensive evaluation, I also replicate the stacked RC simulations of Zhang (2017).

Before delving into the details of the modeling results though, a final note on presentation. In this section, I test *every* metric proposed in Graf et al. (2017) — henceforth, also *original* metrics. Similarly, the following section will evaluate the model across multiple phenomena, multiple syntactic analysis, and a variety of complexity metrics. Thus, to make the results of this chapter more approachable, I assign labels to clusters of metrics as follows:

- ORIGINAL, BASE, RANK = n: all the original base metrics of Graf et al. (2017), as discussed in Chapter 2. No filtered or sorted metrics. RANK = n specifies the cardinality of the ranked metric (e.g., BASE, RANK = 1 only considers unranked base metrics);
- ORIGINAL, FILTERED: The filtered and sorted variants of the original metrics. RANK = n specifies the cardinality of the ranked metric (e.g., FILTERED, RANK = 1 only considers unranked filtered metrics).

[2]Note that Graf et al. (2017) evaluates the MG model on a SRC $<$ ORC preference also for Mandarin Chinese, while Zhang (2017)'s own experiments show an ORC $<$ SRC preference. In general, conflicting evidence as been reported for a subject vs. object preference in Mandarin (Gibson and Wu, 2013, a.o.). Here, I follow what reported in Zhang (2017), as I will rely on her results for the stacked RC cases. However, note that the need to predict ORC $<$ SRC significantly reduces the number of memory metrics that are able to account for the whole set of baseline phenomena. Thus, this choice also results in the most conservative hypothesis.

I then summarize how the whole cluster performs on each processing contrast. Consistently with previous work, I first consider base metrics of ranks 1 and 2. In cases in which base metrics are unsuccessful, I then consider filtered metrics. As most of the original metrics fail on the target preferences, I only discuss the performance of a specific metric in detail when doing so is helpful in shedding light into the functioning of the model. The reader is referred to Chapter 2 for a detailed discussion of these metrics. As a reminder, the load types are summarized in Table 5.2, and shorthands for filters are in Table 5.3.

Load Type	
Tenure(m)	$o(m) - i(m)$
Size(m)	$i(m_i) - i(m_j)$, where m_i is a mover and m_j is its highest target Move node;
Payload(m)	$\lvert\{m \lvert tenure(m) > 2\}\rvert$

Table 5.2: Memory load types for original metrics as defined in Graf et al. (2017).

Filtered Metrics	
M'	M takes intermediate movement steps into account
M_I	M restricted to internal nodes
M_L	M restricted to leaf nodes
M_U	M restricted to unpronounced nodes
M_P	M restricted to pronounced nodes
M^R	applies M recursively

Table 5.3: Notation for filtered metrics as defined in Graf et al. (2017).

Importantly, the preferences for primed RCs seem to parallel Zhang (2017)'s stacked RCs results (see Table 5.1). Since the structural configurations underlying the stacked RCs examples are different than the conjunction of RCs used in the priming cases, it is better to consider those results independently.

5.4.1 Modeling Results: Stacked RCs

Confirming the results reported in Zhang (2017), the MG model as currently defined is not able to account for the parallelism effects found in stacked RCs.

Interestingly though, there is an asymmetry between postnominal and prenominal languages. In particular, if we set aside the Mandarin data *and* adopt a *promotion analysis* of RCs, there *is* in

		OO < *SO*	SS < OS
English	MAXS	Tie	✓
	AVGT	✓	×
Mandarin	MAXS	Tie	Tie
	AVGT	✓	×

Table 5.4: Summary of the performance of ⟨MAXS, AVGT⟩ on staked RCs in Mandarin Chinese and English under a promotion analysis of RCs.

fact a metric able to account for the English stacked RC results together with the baseline results: ⟨MAXS, AVGT⟩.

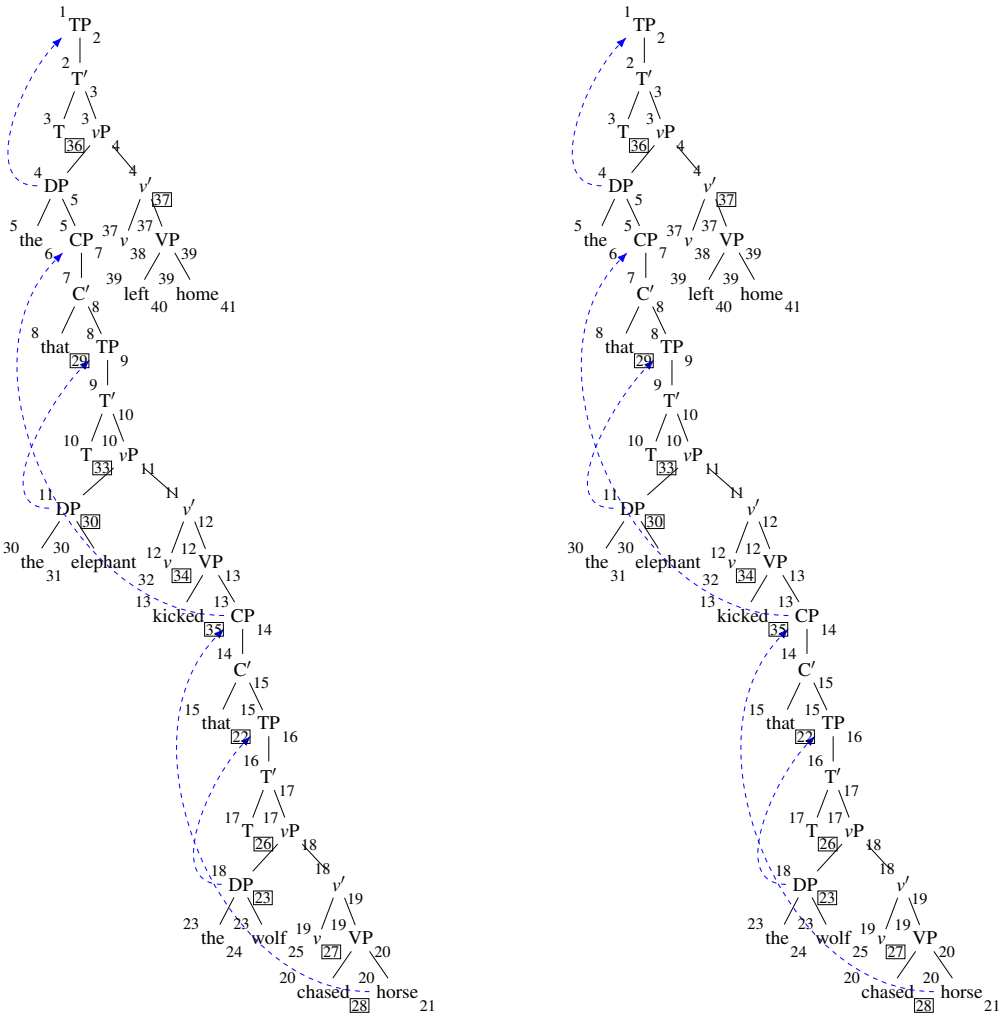
Consider the results in Table 5.4. In English, MAXS is enough to pick up on the *SS* < *OS* contrast $(11-5=6) < (18-11=7)$, see Figure 5.8). This is due to the fact that, in the *OS* case, the ORC is mapped linearly closest to the NP but it has to wait for the object relative to be constructed in the most embedded position. MAXS fails to make the correct predictions in the *OO* < *SO* case, as both constructions tie (MAXS: $13-5=8$) due to the lower ORC raising to the highest Spec,CP position (Figure 5.9). However, the *OO* construction wins on AVGT, as there are overall more nodes with non trivial tenure in the *OO* case.

Things are different for Mandarin. In this case, MAXS also ties in the case of *OO* < *SO* ($20-8=12$ vs $36-24=12$, respectively), and the *OO* construction once again wins on AVGT (Figure 5.11). However, this time MAXS is not able to help in picking up the *SS* < *OS* contrast ($26-14=12 < 36-24=12$) — due to the movement of the lower subject to Spec,RelP in the *SS* construction ($26-14=12$; Figure 5.10).

Importantly, these results *depend* on which syntactic analysis of RCs is picked. Under a wh-movement analysis, the MG predictions for Mandarin Chinese stay unchanged.[3] In English though, MAXS is not able to pick up on the *SS* over *OS* preference (they both tie, Table 5.5). As AVGT predicts *OS* < *SS*, ⟨MAXS, AVGT⟩ fails on this contrast (as does every other RANK = 2 metric).

Finally, we can look at what happens if we also consider metric variants. As it turns out, a recursive version of MAXT can be used to replace MAXS as the highest ranking metric. Then

[3]That is, the overall contrast predicted by the metrics stay unchanged. There are, of course, differences in the numerical values of each metric.

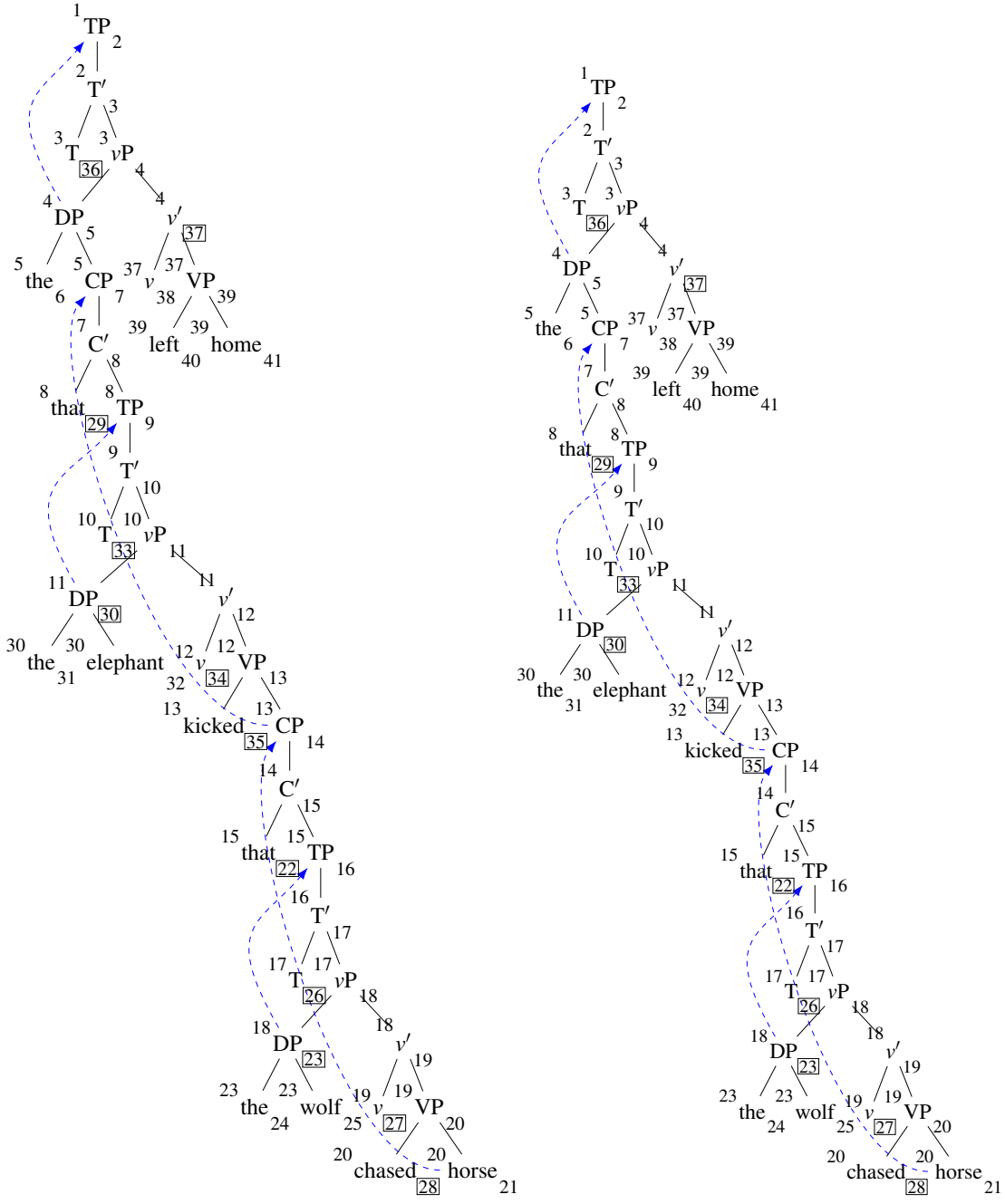


(a) SS (b) OS

Figure 5.8: Annotated English stacked RC (SS vs OS), built following the promotion analysis.

⟨MAXT^R, AVGT⟩ leads to the correct predictions for English stacked RCs under both syntactic analysis (see Table 5.6). However, it is once again unable to account for the correct contrasts in Mandarin Chinese.

Finally, although the metrics' asymmetric behavior across the two languages is interesting, it is important to underline that ⟨MAXS, AVGT⟩ and ⟨MAXT^R, AVGT⟩ are unable to account for all the other baseline phenomena at the same time (see Graf et al., 2017). Thus, the current model remains unsatisfactory in terms of wide empirical coverage.



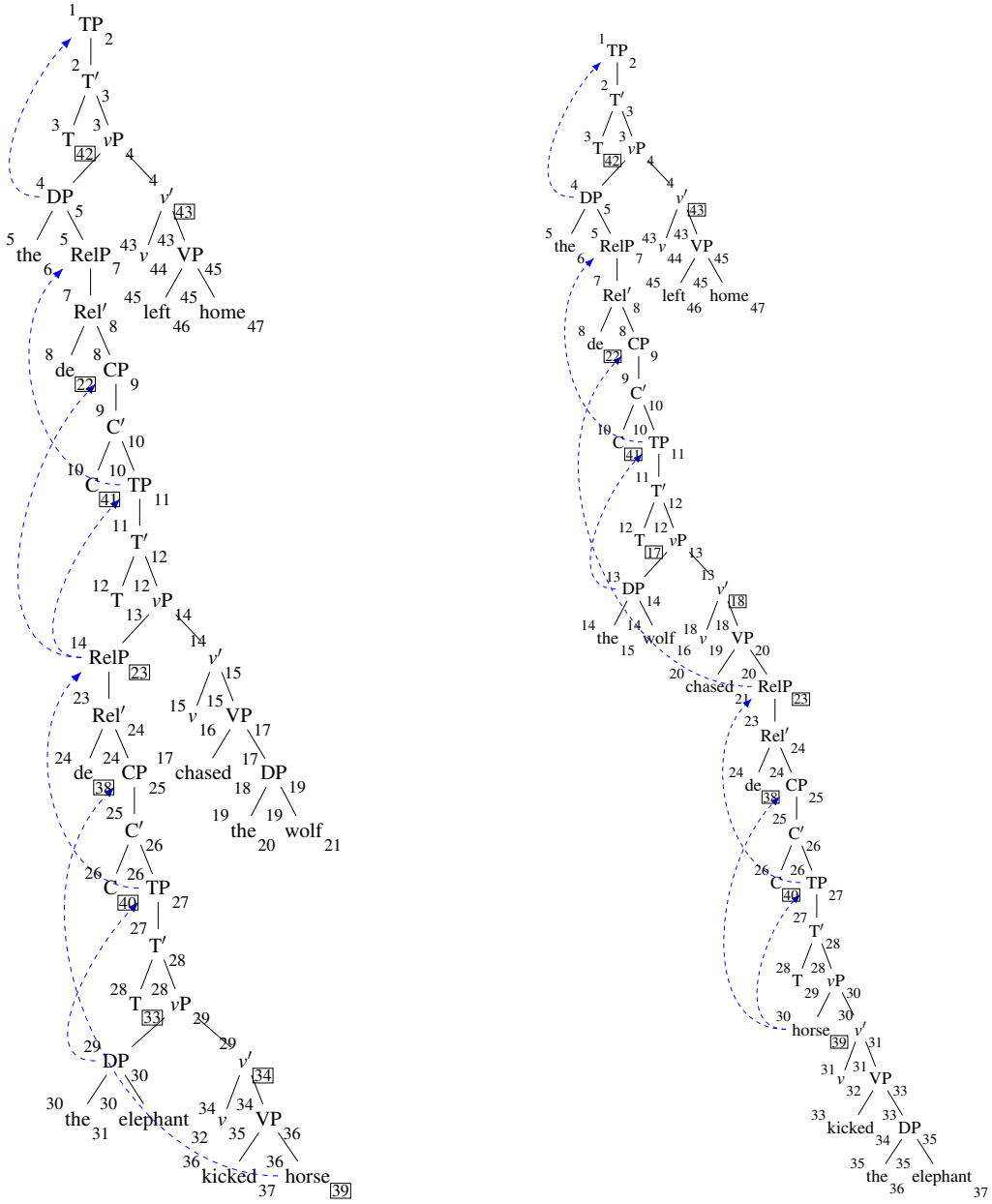
(a) OO (b) SO

Figure 5.9: Annotated English stacked RC (OO vs SO), built following the promotion analysis.

5.4.2 Modeling Results: Priming

Having replicated Zhang (2017)'s results for stacked RCs, we can now turn to priming effects.

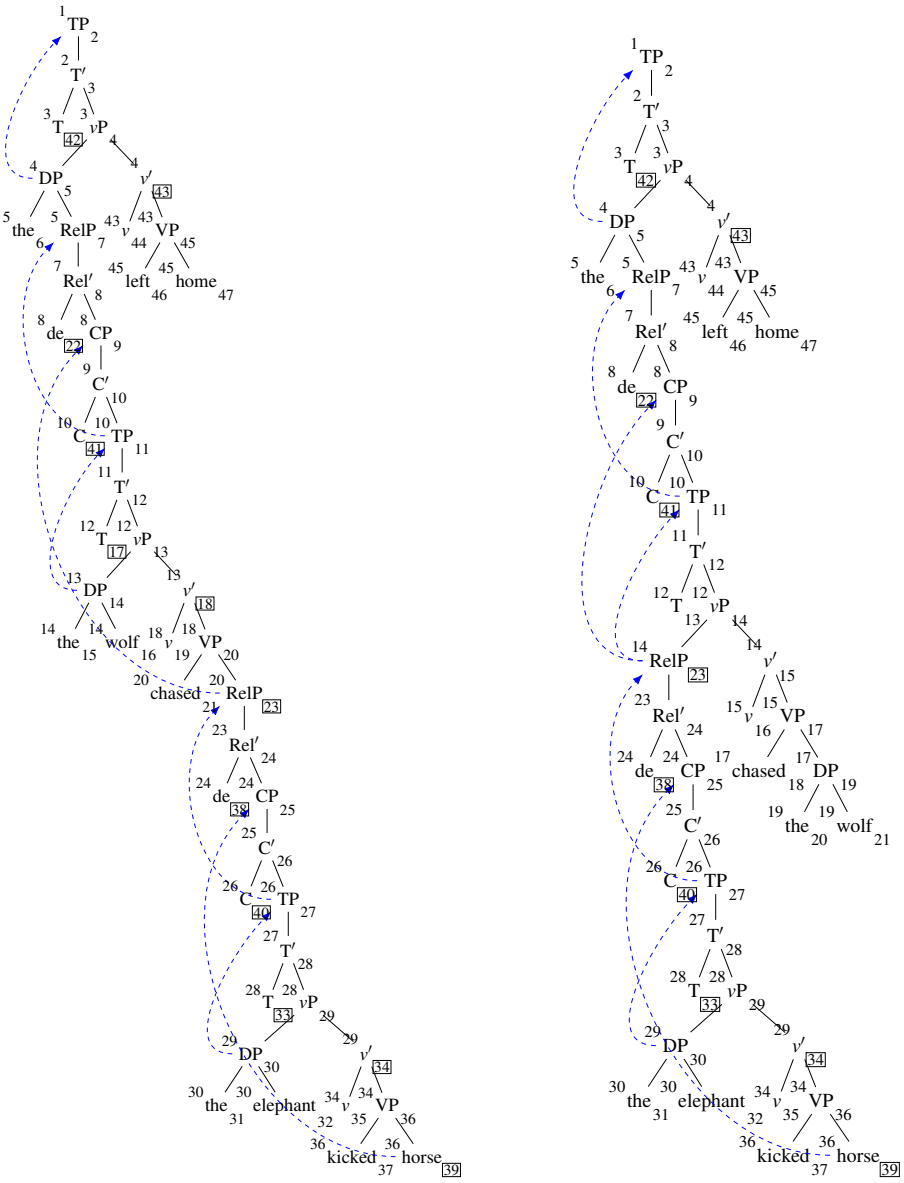
Obviously, adding priming into the set of results will not help with the fact that no existing metric



(a) SS (b) OS

Figure 5.10: Annotated Mandarin stacked RC (SS vs OS), built following the promotion analysis.

can account for both English and Mandarin stacked RCs at the same time. However, it is interesting to compare how the metrics perform on the complexity profiles of primed RCs compared to stacked RCs, as these effects seem to be fundamentally similar.



(a) OO (b) SO

Figure 5.11: Annotated Mandarin stacked RC (OO vs SO), built following the promotion analysis.

Priming alone First, we should look at how the model fares on priming English subject and object RCs by themselves.

Recall that the processing profiles to replicate are those summarized in Table 5.1, and that in this case we are just modeling English RCs. Then, the MG model behaves very similarly to the stacked RC case. Here however, AVGT is enough so capture both the $SS < OS$ and the $OO < SO$ contrast by itself (Table 5.7). Interestingly, these results hold both for the promotion, and for the

		$OO < SO$	SS < OS
English	MAXS	Tie	Tie
	AVGT	✓	×
Mandarin	MAXS	Tie	Tie
	AVGT	✓	×

Table 5.5: Summary of the performance of ⟨MAXS, AVGT⟩ on staked RCs in Mandarin Chinese and English under a wh-movement analysis of RCs.

		$OO < SO$	SS < OS
English	MAXT^R	Tie	✓
	AVGT	✓	×
Mandarin	MAXT^R	Tie	Tie
	AVGT	✓	×

Table 5.6: Summary of the performance of ⟨MAXT^R, AVGT⟩ on staked RCs in Mandarin Chinese and English, under a wh-movement analysis of RCs.

wh-movement analysis of RCs.

Note that in the stacked RC case, AVGT still performs equally across contrasts independently of syntactic analysis. However, for both analyses, it fails to capture the $SS < OS$ contrasts, thus requiring MAXS or MAXT^R to be ranked above it. Given our previous discussion of the stacked RCs' results, while it is somewhat surprising that AVGT is able to predict the correct contrasts for English priming effect, the fact that it remains consistent across syntactic choices is not unexpected.

There are at least two additional things to consider at this point. First, whether any of the current metrics can account for priming while still making the correct predictions for the set of baseline phenomena. Secondly, whether these metrics can account for priming effects *and* for the parallelism effects in the stacked RC cases at the same time, and still derive the right predictions for the baseline effects. The expected answer to both of these questions is no, as we already know that current metrics fail to account for the baseline cases combined with stacked RCs in Mandarin.

Priming + Baseline First, we want to understand whether there are metrics that can account for the primed RCs effects together with the baseline effects. Under a promotion analysis, no BASE metric (either with RANK $= 1$ or RANK $= 2$), is able to capture the whole set of phenomena. However, among the FILTERED, RANK $= 2$ metrics ⟨MAXT$_{IU}$, AVGT$_{IP}$⟩ gives the

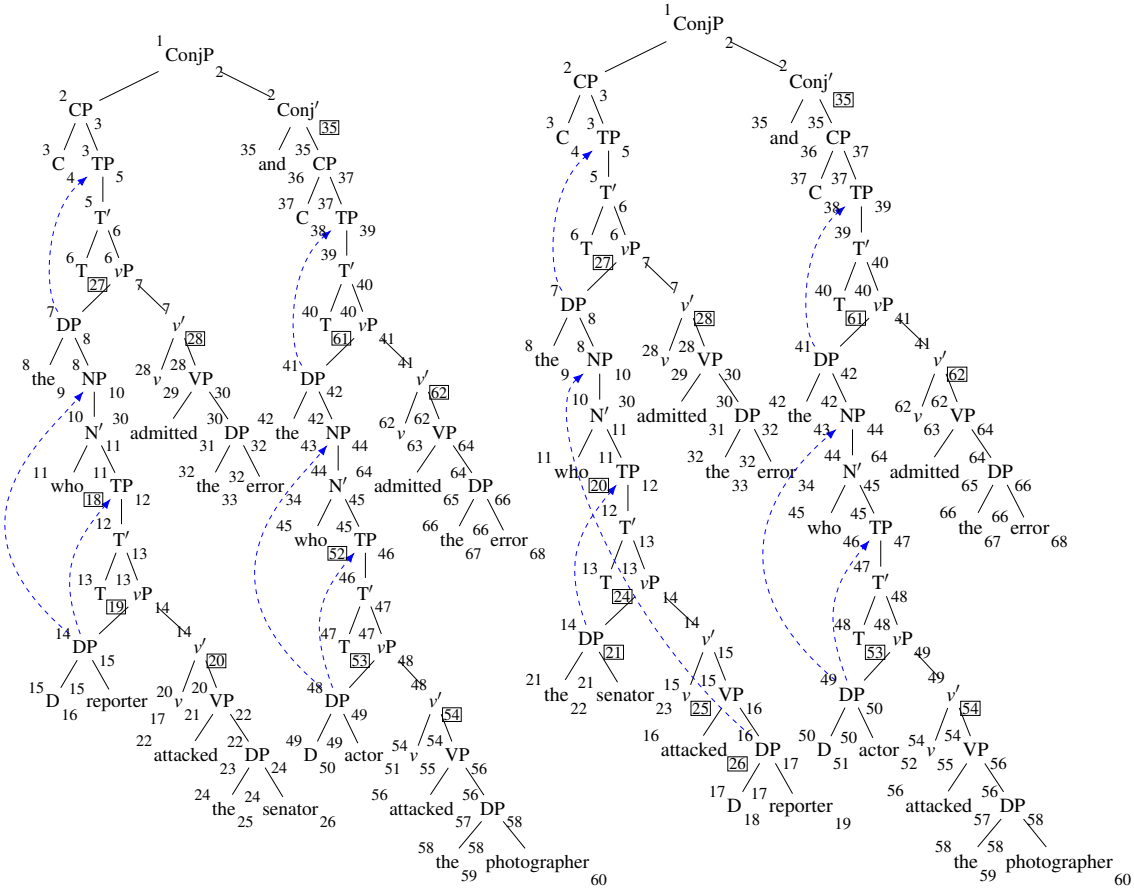


Figure 5.12: Annotated English primed RC (SS vs OS), built following the promotion analysis.

correct predictions: that is, MAXT restricted to internal and unpronounced nodes ranked above AVGT restricted to internal, pronounced nodes. The results are similar with a wh-movement analysis, except that this time the metric $\langle \text{MAXT}_I, \text{AVGT}_U \rangle$ also makes the correct predictions. In both cases, AVGT seems to be playing a crucial role once again.

Priming + Stacked There is no reason to compare to the full set of stacked RCs to the priming cases, as we already know that no metric works on both English and Mandarin stacked RCs at the same time. Let us look instead at the English cases exclusively. In this case, the ranked metric $\langle \text{MAXS}, \text{AVGT} \rangle$ makes the correct predictions with either a promotion analysis of RCs, while $\langle \text{MAXT}^R, \text{AVGT} \rangle$ makes the correct predictions a wh-movement analysis. This is consistent with what discussed above for English stacked and primed RCs under both analyses.

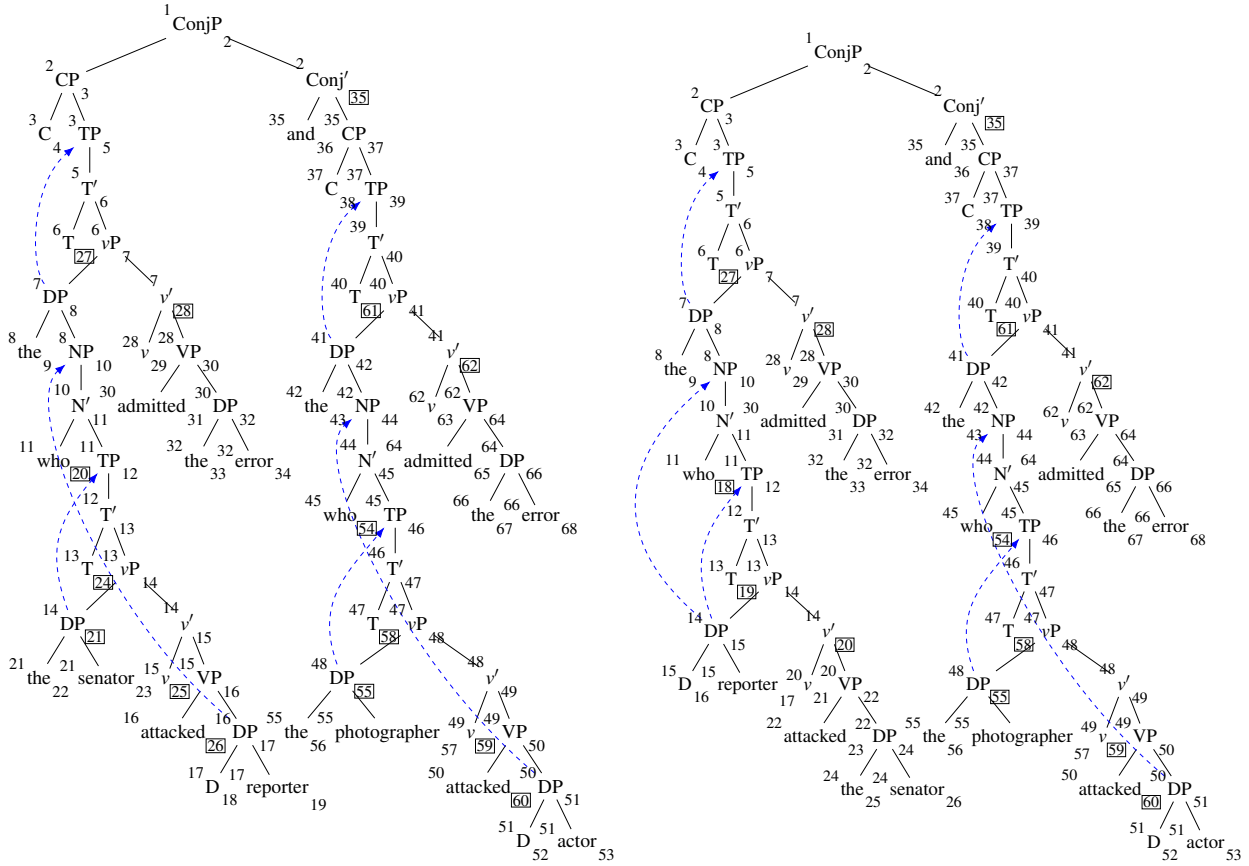


Figure 5.13: Annotated English primed RC (OO vs SO), built following the promotion analysis.

5.4.3 Interim Summary

To sum up the results in this section, AVGT is able to account for the facilitatory effects found in the processing of primed SRCs and ORCs in English. Conceptually, this might seem a surprising results, as the MG model in its current form does not explicitly encode relations between sequences of similar structures. However, this highlights once again how metrics that on the surface appear simplistic — measuring memory just based on the surface geometry of the derivation tree — in fact capture subtle changes in the way the tree traversal strategy impacts memory usage across constructions.[4]

[4]In this sense though, there is an additional point we should make. If we look at the results in Table 5.7 more carefully, we notice that AVGT succeeds on both the $OO < SO$ and the $SS < OS$ only by a margin of decimal points. In previous chapters, we were able to probe the numerical results and clearly correlate them to differences in the underlying syntactic configuration — thus making even small differences across metrics interpretable. However, it is

	Primed RCs		Stacked RCs	
	Wh	Prom	Wh	Prom
OO	12.73	14.06	14.03	15.38
SO	12.86	14.5	14.8	15.63
SS	12.86	14.07	14.8	15.63
OS	13	14.09	15	16

Table 5.7: Summary of AVGT results for primed and stacked RC, modulated by syntactic analysis.

Importantly, putting together what discussed above, it is easy to see that there is no metric that can account for the priming cases, the stacked RCs cases (whether we consider only English, or both English and Mandarin), and the baseline cases together, irrespective of syntactic analysis (Table 5.8). Moreover, there is a fundamental divide in how the metrics perform on primed and stacked cases in English, and in Mandarin Chinese. As our end goal is to provide a model of sentence processing consistent across phenomena *and* languages, the MG approach in its current form remains unsatisfactory.

5.5 Feature Sensitive Metrics & Memory Reactivation

The previous section showed us how metrics like AVGT are able to predict the processing profile of English RC priming to a certain degree. However, it is unclear whether/how these metrics transparently capture the mechanisms underlying priming effects. Importantly, in order to be reproduce the structural parallelism effects that have been argued to lead to facilitatory processing in priming, the MG model model needs to be sensitive to structural repetitions.

As one of the advantages of the MG approach is the interpretability of the linking hypothesis, it seems important to expand the current model, to explicitly encode how building similar structures multiple times can induce facilitatory effects in memory.

As the reader might have noticed, while MGs as introduced in Chapter 2 start as a rich,

unclear what exactly is driving these subtle changes in the average measures. And, in fact, one could argue that most results relying on average are fundamentally dependent on rounding decisions. For instance, most of the differences reported in this section would disappear if we were to round the metrics values up. While in what follows I will treat average-based claims with skepticism, it is important to note that the average values are rounded here to the second decimal point just for ease of presentation. Unrounded values are actually used in the underlying comparisons.

		Metrics	Success?
Stacked	English	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	✓
		ORIGINAL, FILTERED, RANK = 2	✓
	Mandarin	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	×
		ORIGINAL, FILTERED, RANK = 2	×
Stacked + Baseline	English	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	×
		ORIGINAL, FILTERED, RANK = 2	×
	Mandarin	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	×
		ORIGINAL, FILTERED, RANK = 2	×
Priming		ORIGINAL, BASE, RANK = 1	✓
		ORIGINAL, BASE, RANK = 2	✓
		ORIGINAL, FILTERED, RANK = 2	✓
Priming + Stacked	English	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	✓
		ORIGINAL, FILTERED, RANK = 2	✓
	Mandarin	ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	×
		ORIGINAL, FILTERED, RANK = 2	×
Priming + Baseline		ORIGINAL, BASE, RANK = 1	×
		ORIGINAL, BASE, RANK = 2	×
		ORIGINAL, FILTERED, RANK = 2	✓

Table 5.8: Summary of the performance of each cluster of current MG metrics, over sets of processing phenomena.

lexicalized encoding of current minimalist analyses, the specific implementation of the parser used in our processing model adopts a much simpler representation of syntactic structures, which discards the feature component of the lexical items.

The existing literature on MG parsing has consciously adopted metrics that ignore the feature-based component of MG trees, and focus exclusively on geometrical relationships defined over the derivation tree. However, by doing so the model ends up ignoring important information about a sentence's derivation, as the set of features driving the derivation constitutes the real core of modern minimalist approaches to syntactic structure (Adger, 2003). This choice arguably undermines the formalism's tight connection to modern analyses of syntactic phenomena. Moreover, the psycholinguistic literature on sentence processing has independently shown that syntactic features matter towards processing complexity (Rogalsky and Hickok, 2008; Parker et al., 2017; Lewis and Vasishth, 2005, a.o.).

Recall that in MGs the features carried by a lexical item express all the information needed to reconstruct that item's argument structure. In fact, MG features explicitly capture the sequence of Merge and Move operations that the parser has to resolve to build a syntactic derivation.

Thus, it should be possible to make the parser aware of structural operations that have already taken place, by making it sensitive to feature configurations. This seems to be in line with Zhang (2017)'s intuition, that the existing MG metrics fail to capture the processing profile of stacked RCs, specifically because of their inability to account for syntactic features. Grounded in this idea, we can look once again at the priming literature to build psychologically plausible, feature-based variants of the existing memory metrics.

Existing computational approaches to syntactic priming include implicit learning models (Bock et al., 2007); activation-based accounts (Reitter et al., 2011); and hybrid models (Jaeger and Snider, 2013). These accounts are based on different assumptions about the nature of the mechanisms underlying priming:

1. *residual memory activation* of a previously encountered syntactic structure, leading to short-term priming effects (Pickering and Branigan, 1998);

2. long-term *implicit learning effects*: the unconscious acquisition of abstract information

processing routines (Bock and Griffin, 2000; Bock et al., 2007).

The debate about which of these accounts is better supported by experimental evidence is still open. Crucially though, residual activation approaches give us a way to investigate priming phenomena with an MG parser, by reintegrating features into the MG derivations and introducing metric measuring *feature reactivation*.

Importantly, the focus on memory activation processes is not to be interpreted as a stance against implicit learning accounts. However, it is in line with the choice to ignore the effects of probabilistic information on processing difficulty — and, in fact, to put aside all other factors apart from purely structural ones.

This section explores in detail the different ways complexity metrics can be made to take syntactic features into account. In particular, I will formalize and extensively evaluate a set of metrics measuring a notion of reactivation, as associated to movement features.

5.5.1 Encoding Feature Reactivation

The first question to ask is, of course, how to encode reactivation in the MG parser. Here, I will follow a procedure based on how the parser keeps track of movers (Zhang, 2017).

Intuitively, movers are stored by the MG parser in specific memory cells, each dedicated to a particular movement type (i.e., feature). Movers triggered by the same feature are thus stored in the same cell. If a memory cell has been inactive for a long time, storing a mover in that cell comes with a certain activation cost. However, if that memory cell has recently stored another mover, putting the next mover into it should be less costly.

The procedure described above can be implemented by counting the number of parsing steps between movements of the same type, thus effectively accounting for the derivational time between two movers. Given Kobele et al. (2013)'s annotation schema, *reactivation* is computed by subtracting the outdex of a movement node from the index of the next one.

REACTIVATION For each node m_i associated to a movement feature f^-, its reactivation is $i(m_i) - o(m_{i-1})$; the index of m_i minus the outdex of the closest preceding node also associated to f^-, if it exists.

This definition of reactivation essentially indexes how costly it is to store some kind of movers compared to others. In practice, this can be interpreted as the *encoding* cost associated to a node.

Note, however, that reactivation is supposed to encode a facilitatory effects induced by structural repetition. According to the definition above though, there is no reactivation value assigned to movement features that appear for the first time. This might lead to issues in our comparative approach, as derivations without movement repetitions would have non-existent reactivation values — and thus, counterintuitively, might be evaluated as recruiting fewer memory resources. To account for that, the previous definition describes what I will refer to as PLAIN REACTIVATION:

$$\mathrm{R}_p(m_i) \qquad i(m_i) - o(m_{i-1})$$

Then, REACTIVATION is computed instead as:

$$\mathrm{R}(m_i) \quad \begin{cases} 1 - \frac{1}{R_p}, \text{ if } \exists o(m) \wedge R_p > 0 \\ 0, \text{ if } R_p \leq 0 \\ 1, \text{ if } \neg \exists o(m_{i-1}) \end{cases}$$

Essentially, R associates a *default* reactivation value to each node (1), which can then be reduced in case of facilitatory effects (for practical reasons, R is also given a lower bound equal to 0).

The idea that encoding something in memory comes at a cost has been well motivated by psycholinguistic insights (Van Dyke and McElree, 2006; McElree et al., 2003; McElree, 2006; Lewis et al., 2006; Villata et al., 2018), but has been lacking from the definitions of memory usage adopted by the MG model. Thus, feature reactivation metrics building on this idea might also bring the MG model notion of memory usage closer to that of other, cognitively grounded, sentence processing frameworks.

5.5.1.1 Base Metrics

As for tenure and size, reactivation is just a measure over a node, but it cannot be directly used to compare derivations. However, it is possible to define multiple metrics that use this concept to compute reactivation values over a full derivation. Let $\mathscr{M}$ be the set of movement features in a

derivation for a tree T, and M^f a tuple containing all nodes m associated to a feature $f^- \in \mathscr{M}$, ordered based on their linear precedence in the string yield in T. Then we define:

$\textsc{SumR}^f \quad \sum_{m_i \in M^f} R(m_i)$

$\textsc{MaxR}^f \quad max(\{R(m_i) | m_i \in M^f\})$

$\textsc{AvgR}^f \quad \frac{\textsc{SumR}^f}{|M^f|}$

Note that these metrics are all defined with respect to the reactivation of nodes associated to one specific movement feature. However, it makes more sense to evaluate the overall effects of every reactivated node during the parse:

$\textsc{SumR} \quad \sum_{f \in \mathscr{M}} \textsc{SumR}^f$

$\textsc{MaxR} \quad max(\{\textsc{MaxR}^f | f \in \mathscr{M}\})$

$\textsc{AvgR} \quad \frac{\textsc{SumR}}{|\mathscr{M}|}$

Obviously, values for these metrics will significantly depend on whether the set of movers M^f includes intermediate landing sites or not. As already observed, intermediate landing sites do not affect the tree-traversal strategy of Stabler's parser, thus it might make sense to factor them out when measuring reactivation. Modeling results comparing the predictions of these two kinds of metrics will also offer insights into the relevance of intermediate landing sites to derivational theories of syntax.

5.5.1.2 Modeling Interactions

Once base reactivation metrics are defined, what remains to be explored is how they interact with the original storage-based metrics previously explored in the literature. Obviously, the most immediate step is to evaluate ranked metrics incorporating new and original metrics. However, there are other ways in which reactivation can interact with storage.

To see how, note that REACTIVATION as defined above is different from the idea of memory activation usually adopted in the psycholinguistic literature, tightly linked to the notion of *decay*

(Van Dyke and Lewis, 2003). Decay is a fundamental concept in psychologically grounded memory-burden models. As a node is stored in memory, it is associated with a certain activation level. Then, the longer it stays in memory without being accessed, the lower its activation level gets (it decays), leading to increased efforts when said node finally needs to be retrieved.

Since tenure and size are both in some ways compatible with the idea of decay (Kobele et al., 2013), the insight that reactivation should directly impact the cost of maintaining a node in memory can be formalized in metrics that weight tenure and size of a node based on its feature reactivation. I implement this idea as follows.

For each node m associated to a movement feature f, its SIZE BOOST is:

BS $\quad$ $\textsc{Size}(m) * \textsc{Reactivation}(m)$

Similalry, its TENURE BOOST is:

BT $\quad$ $\textsc{Tenure}(m) * \textsc{Reactivation}(m)$

As a variant of tenure, BOOST formalizes the reactivation effect on the memory burden associated to keeping a node in memory for a long time. As a variant of size, BOOST can be interpreted as encoding how some features weight less in memory due to an increase in their activation level.

Finally, we can define a metric that weights the combined effects of reactivation on size and tenure:

BTS $\quad$ $\textsc{Tenure}(m) * \textsc{Size}(m) * \textsc{Reactivation}(m)$

Note again that, since reactivation is meant to encode a facilitatory effect, values for R, BS, BT and BTS will decrease as the number of similar movement dependencies in a derivation increases. Then, these notions are implemented in metrics like SUMB, MAXB, and AVGB, computed exactly as the mirroring ones defined for size and tenure. Moreover, tenure based metrics are also associated to their filtered variants.

5.6 Memory Reactivation: Model Evaluation

As in Section 5.4, I will look for metrics that are successful on priming and stacked RC effects. Additionally, we need to carefully explore how the new metrics fare on baseline phenomena by themselves. Consistently with what was done in the previous section — and in order to make the discussion of the modeling results approachable — I assign labels to clusters of metrics as follows:

- REACTIVATION, BASE, RANK = n: just reactivation metrics, unfiltered. RANK = n specifies the cardinality of the ranked metric;
- REACTIVATION, FILTERED, RANK = n: reactivation metrics, with and without filters;
- REACTIVATION, FULL, RANK = n: original metrics (as in Section 5.4) and reactivation metrics, filtered and unfiltered.

For each processing contrast, I then summarize how the whole cluster performs. A summary of the new, reactivation-based memory load types defined in the previous section is presented in Table 5.9

Load Type	
Rp	$i(m_i) - o(m_{i-1})$
R	$\begin{cases} 1 - \frac{1}{R_p}, \text{ if } \exists o(m) \wedge R_p > 0 \\ 0, \text{ if } R_p \leq 0 \\ 1, \text{ if } \neg\exists o(m_{i-1}) \end{cases}$
BT	$\text{TENURE}(m) * \text{R}(m)$
BS	$\text{SIZE}(m) * \text{R}\ (m)$
BTS	$\text{TENURE}(m) * \text{SIZE}(m) * \text{R}(m)$

Table 5.9: Memory load types for reactivation metrics as defined in Section 5.5.1.

5.6.1 Modeling Choices: Feature Selection

As mentioned, this chapter focused on two distinct analyses for RC constructions so to take the MG parser's sensitivity to minor structural differences into account. Thus, as pointed out before, syntactic choices are a crucial degree of freedom for the modeling approach.

Extending the model with metrics sensitive to feature reactivation complicates this picture even further. Now, it is not just differences in the geometry of the derivation trees which might affect the results of the model. Instead, we are going to have to commit to a set of features consistent with the derivational operations posited by each analysis.

Modern generative syntax, and the Minimalist Program in particular, presupposes features as a fundamental component of its syntactic objects (Chomsky, 1995; Adger, 2003; Collins and Stabler, 2016). While there have been attempts in the literature to formulate a complete theory of syntactic structures that fully relies on an explicit, feature-based machinery (Rizzi, 1990; Adger, 2003; Friedmann et al., 2009) though, there is still widespread disagreement on what a correct theory of syntactic features should look like (den Dikken, 2000; Adger and Svenonius, 2001, a.o.).

Moreover, we have to be careful not to mix the specific implementational details of the grammar formalism used as the backbone of the parser (MGs), with the assumptions of the syntactic theory under consideration (Minimalism). As Stabler (2013) points out, it is possible to assign features to MGs' lexical items in a way that allows us to formalize many proposals in the syntactic literature. However, this does not guarantee that the feature choices forced by the formalism will be motivated on syntactic grounds. For instance, each theoretical analysis can be implemented in MG with a plethora of alternative featural configurations.

As this chapter constitutes a first exploration of feature-based MG metrics, it seems reasonable to be as conservative as possible in the choice of feature overlaps. Specifically, since reactivation metrics are restricted to movement operations, I will only focus on whether particular movement steps could reasonably be triggered by the same feature, or not. When in doubt — i.e., in cases in which both interpretations are consistent with what discussed in the literature — I will assume that movement was triggered by distinct features.

Moreover, in what follows I consider features just as elements of the formalism triggering movement operations. I assign them names that, when possible, are consistent with theoretical assumptions about the nature of each movement operation. However, I do not make any specific commitment to the ontological status of each feature within an overall syntactic theory.

Consider Figure 5.14: a derivation for one of the English primed RC test cases (the *OO* one, as in (42)), under a promotion analysis. Here, the DP containing the RC moves to fill the subject

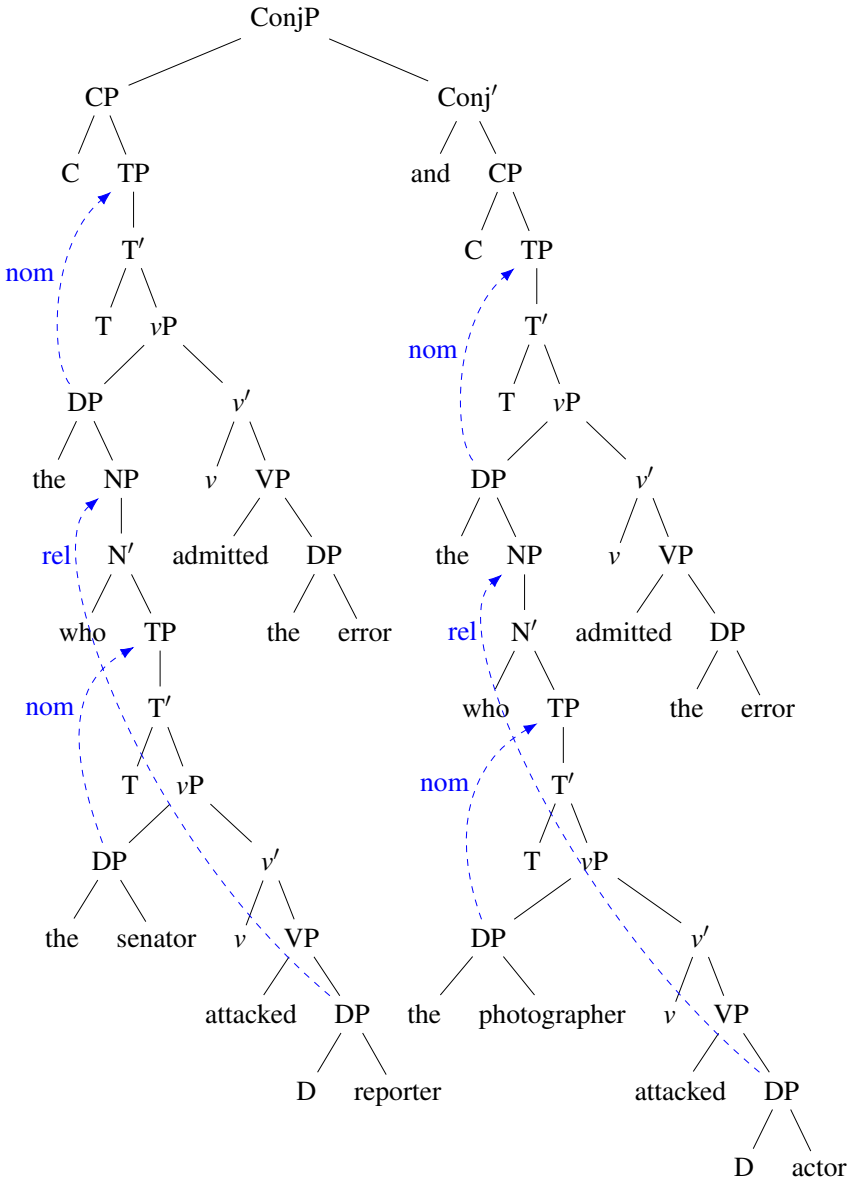


Figure 5.14: Feature choices for English primed RCs (promotion analysis, OO).

position in Spec,TP of the main clause. In this chapter's simulations, this movement is associated to a *nom* feature. This choice is consistent with the widespread idea that English subjects usually bear nominative, and that nominative is assigned by T. Note however, that in current versions of minimalism case assignment does not need to be local, and thus it is not immediately obvious that movement of the DP should be triggered by a *nom* feature. For our purposes though, it is enough to mark that the feature that triggers this movement in the main clause is probably the same that triggers the DP to Spec,TP movement inside the RC. For similar reasons, the movement of the relativized object from its base position to Spec,CP of the RC is simply labeled with a *rel* feature.

Consider then the stacked RC cases for Mandarin and English (Figure 5.15a and Figure 5.15b, respectively). The feature choices here are basically the same as in the primed case. However, some decision needs to be made regarding the remnant movement steps in the Mandarin case (TP to RelP). As such movement operations are needed by Kayne's analysis in order to derive the correct preverbal word order, they are simply marked with a *wo* (word order) feature. Again, this is not an ontological claim about the status of word order features in our syntactic theories — I am merely concerned with highlighting the difference between this remnant movement step, and the relativization one. This choice is then consistent with any theory of features that assumes a distinctions between the triggers for these movement operations.[5]

5.6.2 Modeling Results: Baseline Phenomena

Technical definitions for reactivation-based metrics in place, and modeling choices clarified, the first thing to do it to test how such metrics perform over the baseline results.

Independently on which analysis of RCs is picked, no REACTIVATION, BASE, RANK $= 1$ is able to account for the whole set of baseline phenomena. The closest we come to success is with MAXR and MAXR$'$, which tie on every construction.[6]

If we ignore the center-embedding and the SC-RC/RC-SC cases, it is interesting to observe

[5]Similar choices had to be made for all the test cases considered in this chapter. That is, I annotated derivations for all the baseline cases, the stacked RC cases, and the primed RC cases. Trees for these choices are omitted here but are available, together with the code for the reactivation metrics, at https://github.com/aniellodesanto/mgproc_reactivation.

[6]Recall that "prime" variants — as MAXR$'$ with respect to MAXR — are a version of the original metric which take intermediate movement steps into account.

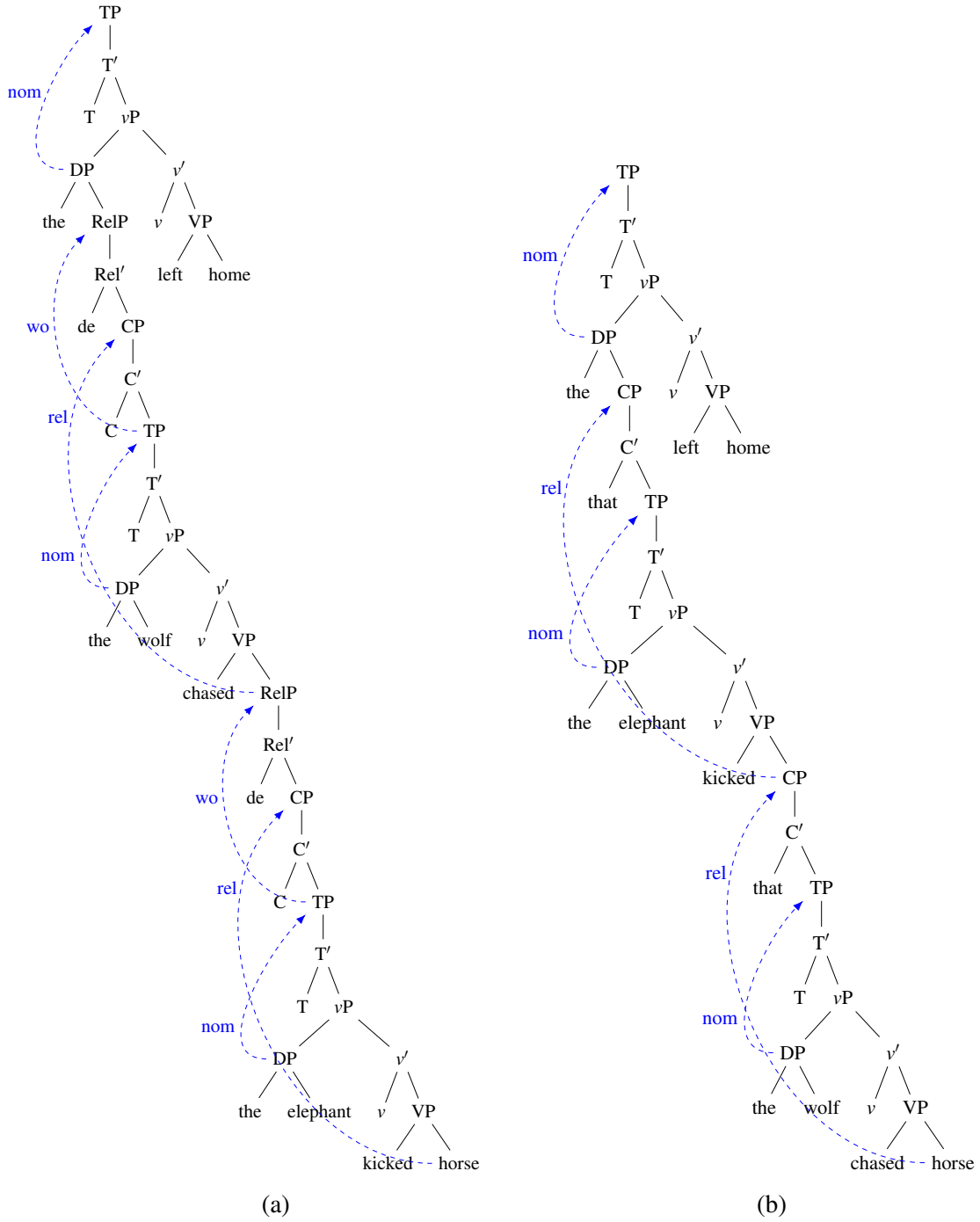


Figure 5.15: Feature choices for Mandarin (a) and English (b) stacked RCs (promotion analysis, OO).

that there is a wide set of reactivation metrics making the right predictions for the SRC vs ORC contrasts. Even so, no individual metric is able to account for every RCs contrast cross-linguistically at the same time. There also doesn't seem to be a clear pattern in terms of structural differences across languages — as Mandarin RCs are covered by the same metrics as English, while Japanese and Korean require different ones.

Importantly, no reactivation metric is able to account for the English right embedding $<$ center embedding contrast. In fact, with the exception of MAXR (which predicts a tie), most metrics predict the opposite processing profile. These results hold also for REACTIVATION, RANK $= 2$ metrics, whether we use filters or not.

Unsurprisingly, we get more coverage if we consider the full set of RANK $= 2$ metrics. In particular, a variety of metrics ranking AVGT or MAXT first correctly account for the full set of baseline phenomena — consistently with the fact that these two metrics predict a majority of the baseline phenomena by themselves.

Crucially, the baseline phenomena considered here have not been directly associated to memory reactivation effects in the psycholinguistic literature. Thus, it should not be surprising that most of the new metrics are not be able to reproduce every processing asymmetry in this set. However, there are repetitions of movement dependencies in these constructions, that are picked up by the reactivation metrics, and it is interesting that a number of rankings of original metrics and new ones can give us the correct predictions. What matters then, is whether it is possible to define an appropriate combination of original and reactivation-based MG metrics that could work for the baseline phenomena, the stacked RC phenomena, and the primed RC cases at the same time.

5.6.3 Modeling Results: Stacked RCs

We can now look at how reactivation-based metrics perform on the processing of stacked RC. As feature reactivation was partially inspired by the parallelism effects in the stacked cases, it might be reasonable to expect widespread success of the enriched model on the test cases. However, that is not the case.

Stacked RCs None of the REACTIVATION, BASE, RANK $= 1$ metrics is able to account for the stacked RC results. Once again, the problem seems to be the inability of the metrics to account for the English and Mandarin contrasts at the same time.

More precisely, a majority of the new metrics predicts the English processing profile correctly. Most of the same metrics also successfully account for the $OO < SO$ contrast in Mandarin. What these metrics are unable to capture is the Mandarin $SS < OS$ asymmetry. A few metrics — for instance, MAXR^R, SUMBS, SUMR — *do* predict $SS < OS$ for Mandarin, but then get the other contrasts wrong. As before, the closest we come to a satisfactory results is with MAXR, which ties on every contrast.

Things improve when we start looking at RANK $= 2$ metrics, with *and* without filters. With a promotion analysis, the correct contrasts are predicted by a selection of metrics ranking MAXR$'_p$ first. For instance, Table 5.10 and Table 5.11 detail the performance of $\langle$MAXR$'_p$, AVGBT$\rangle$ and $\langle$MAXR$'_p$, AVGBTS$\rangle$ for English and Mandarin.

	English		**Mandarin**	
	$OO < SO$	$SS < OS$	$OO < SO$	$SS < OS$
$\langle$MAXR$'_p$, AVGBT$\rangle$	✓	✓	✓	✓
$\langle$MAXR$'_p$, AVGBTS$\rangle$	✓	✓	✓	✓

Table 5.10: Stacked RCs (promotion analysis): Successful reactivation metrics

There are a few things that are worth pointing out. First of all, AVGBT and AVGBTS make the correct prediction on every contrast, except the $SS < OS$ case in Mandarin. As mentioned above, this is in fact the case for a majority of reactivation metrics. MAXR$'_p$ helps with this, as it ties on every contrast except the one missed by the boost based metrics.

Secondly, I pointed out before some conceptual issues with the way plain reactivation (R$_p$) is defined, introducing the more controlled notion of reactivation indexed by R. In general, R and R$_p$ behave very similarly. However, MAXR$'$ ties on the Mandarin $SS < OS$ contrast (see Table 5.12). Moreover, note that the correct results depend on the *prime* variant MAXR$'_p$, and in fact MAXR$_p$ leads to predicting a tie in Mandarin. Thus, differently than what observed in previous work, it looks like intermediate movement steps significantly contribute to deriving the correct processing profiles.

		MaxR$'_p$	AvgBT	AvgBTS
English	$OO < SO$	Tie	✓	✓
	$SS < OS$	Tie	✓	✓
Mandarin	$OO < SO$	tie	✓	✓
	$SS < OS$	✓	×	×

Table 5.11: Stacked RCs (promotion analysis): Successful reactivation metrics decomposed

	English		**Mandarin**	
	$OO < SO$	$SS < OS$	$OO < SO$	$SS < OS$
MaxR$'$	✓	✓	✓	✓
MaxR$'_p$	✓	✓	✓	✓

Table 5.12: Stacked RCs (promotion analysis): Comparing the performance of MaxR$'$ and MaxR$'_p$

Interestingly, no metric ranking MaxR$'_p$ first succeeds when using a wh-movement analysis of RCs. Specifically, undert this analysis MaxR$'_p$ fails on the English $SS < OS$ contrast, predicting the opposite processing preference. All successful metrics under the wh-movement analysis have one of the max, boost-based metrics as their highest metric, and some recursive variant of R or R$_p$ ranked lowest.

For instance, Table 5.13 illustrates the performance of $\langle$MaxBT, MaxR$^R_p\rangle$. As these results show, MaxBT makes the correct predictions for English, but fails to account for the $SS < OS$ contrast in Mandarin. MaxR^R_p is then necessary to discriminate in that case. Note that the recursive variant of MaxR$_p$ is necessary, as MaxR$_p$ by itself also predicts a tie on this contrast.

	Stacked English		**Stacked Mandarin**	
	MaxBT	MaxR^R_p	MaxBT	MaxR^R_p
OO	4.5	[14,10,9]	2	[26,18,17,17]
SO	4.79	[24,16]	2	[26,24,17]
SS	1	[14]	7.70	[27,24,17]
OS	4.44	[12,9]	7.70	[27,25,17]

Table 5.13: Success of $\langle$MaxBT, MaxR$^R_p\rangle$ on stacked and primed RCs

Finally, and not surprisingly, the amount of metrics that is able to account for stacked RCs increases, if we include the original metrics into the ranking. In particular, numerous metrics ranking MaxT above reactivation-based metrics make the correct predictions both in English and

in Mandarin. Strikingly though, there still doesn't seem to be a metric that works the same way across both analyses of RCs.

We can now look at what happens when also considering the baseline cases.

Stacked + Baseline Although on the stacked RC cases the reactivation metrics are less successful than what we might have expected, it was still possible to point out a few metrics making the correct predictions for English and Mandarin both.

The fact that metrics ranking AVGT and MAXT first were successful is also encouraging, as we know from the previous section that MAXT and AVGT make good predictions on the baseline cases. However, it turns out that, when using a promotion analysis, none of the metrics correctly accounting for the stacked RC results is also able to make the right predictions for the baseline cases (see Table 5.14).

	Stacked RCs		**SRC < ORC**			**ORC < SRC**	**SC/RC < RC/SC**	**Right < Center Embedding**
	English	Mandarin	English	Korean	Japanese	Mandarin	English	English
$\langle \text{MAXR}'_p, \text{AVGBT} \rangle$	✓	✓	✓	×	×	✓	tie	×
$\langle \text{MAXR}'_p, \text{AVGBTS} \rangle$	✓	✓	✓	×	×	✓	tie	×

Table 5.14: Performance of $\langle \text{MAXR}'_p, \text{AVGBT} \rangle$ and $\langle \text{MAXR}'_p, \text{AVGBTS} \rangle$ on stacked RCs and baseline phenomena, under a promotion analysis of RCs.

Things are once again slightly better under a wh-movement analysis. In this case, there are a few filtered metrics ranking MAXT first (e.g., $\langle \text{MAXT}_{IU}, \text{AVGR} \rangle$ and $\langle \text{MAXT}_{IU}, \text{AVGR}_p \rangle$), which make the correct predictions across the board. This is interesting, as previous chapters already noted how MAXT is grounded in psychologically plausible ideas about memory usage, and its values can be easily reconstructed by looking at the derivation tree. Thus, it would be an exciting result if metrics like $\langle \text{MAXT}_{IU}, \text{AVGR} \rangle$ could take the comprehensive role fulfilled by $\langle \text{MAXT}_{IU}, \text{SUMS} \rangle$ in previous work.

Then, what remains to be done is test these metrics on the priming cases.

5.6.4 Modeling Results: Priming

At this point, we have a general understanding on how reactivation-based metrics work for cases that involve more explicit (stacked RCs) and less explicit (baseline) instances of parallel movement operations. To complete our evaluation, we can finally test the model on examples of priming effects: primed subject and object RCs.

Priming alone When evaluating primed RCs by themselves, the reactivation metrics are strikingly successful. In particular, with a promotion analysis, even RANK = 1 metrics — namely, AVGR′, AVGBS, and SUMR — lead to the correct predictions. This is in clear contrast with the modeling results on the stacked RC examples. None of the RANK = 1 metrics is successful under a wh-movement analysis (cf. Table 5.15).

	Promotion		**Wh-Movement**	
	OO < SO	SS < OS	OO < SO	SS < OS
AVGR′	✓	✓	✓	×
AVGBS	✓	✓	×	✓
SUMR	✓	✓	×	✓

Table 5.15: Performance of AVGR′, AVGBS, and SUMR on the primed RC contrasts.

However, a good number of RANK = 2, unfiltered reactivation metrics is successful for both analyses. Specifically, most winning metrics rank boost metrics (e.g., MAXBT, AVGBTS) first. Once again, adding filters of different kind doesn't contribute much to the winning metrics.

Finally, a significant number of RANK = 2 metrics mixing original MG metrics and reactivation ones is also successful on the priming cases. Importantly, $\langle \text{MAXT}_{IU}, \text{AVGR} \rangle$ — one of the few metrics also successful on the Stacked RC cases and all the baseline cases — makes the right predictions for RC priming under both syntactic analyses.

Priming + Stacked While numerous reactivation metrics are successful on the priming cases, and some metrics correctly account for the stacked RC cases, the results when considering both phenomena are overall disappointing.

In particular, with a promotion analysis of RCs, no metric is able to account for the priming

preferences together with the stacked RC preferences (not even $\langle$MAXT$_{IU}$, AVGR$\rangle$, see Table 5.20). This is puzzling, since primed RCs and stacked RCs seem to be involving similar structural configurations (and in fact, early in the chapter we suggested that stacked RCs could be considered as a subcase of a more general priming phenomenon).

	OO $<$ SO	SS $<$ OS
$\langle$MAXR$'_p$, AVGBT$\rangle$	✓	×
$\langle$MAXR$'_p$, AVGBTS$\rangle$	✓	×

Table 5.16: Performance of $\langle$MAXR$'_p$, AVGBT$\rangle$ and $\langle$MAXR$'_p$, AVGBTS$\rangle$ on primed RCs (under a promotion analysis).

		MAXR$'_p$	AVGBT	AVGBTS
English	$OO < SO$	Tie	✓	✓
Stacked RCs	$SS < OS$	Tie	✓	✓
Mandarin	$OO < SO$	Tie	✓	✓
Stacked RCs	$SS < OS$	×	×	×
English	$OO < SO$	✓	×	×
Primed RCs	$SS < OS$	×	✓	✓

Table 5.17: Individual performance of MAXR$'_p$, AVGBT , and AVGBTS on stacked and primed RCs (under a promotion analysis).

As just a few reactivation metrics are successful on the stacked RC contrasts under a promotion analysis, it can be informative to look only at those and compare their performance on the primed RC cases.

In particular, as summarized in Table 5.16 and Table 5.16, AVGBT and AVGBTS are successful on the $SS < SO$ contrast for English in the stacked case, but fail in the primed case (see also Table 5.18 for a numerical decomposition of these results).

	Mandarin Stacked			**English Stacked**			**English Primed**		
	MAXR$'_p$	AVGBT	AVGBTS	MAXR$'_p$	AVGBT	AVGBTS	MAXR$'_p$	AVGBT	AVGBTS
OO	12	1.50	12.60	12	3.73	12	32	2.59	9.75
SO	12	2.61	18.21	12	4.45	14.62	33	1.950	8.01
SS	6	4.11	21.55	6	0.96	3.9	35	0.98	4.92
OS	6	2.54	14.32	6	1.84	7.11	33	1.954	7.83

Table 5.18: MAXR$'_p$, AVGBT, and AVGBTS values for stacked and primed RCs

Finally, under a wh-movement analysis, there are metrics that can account both for the primed and stacked RC cases — for instance, Table 5.19 shows the prediction of the metric $\langle$MAXBT, MAXR$'^{R}_{p}\rangle$, already discussed above for the stacked RC cases. More importantly, $\langle$MAXT$_{IU}$, AVGR$\rangle$ again derives the correct results across the board (Table 5.20).

	English Stacked		**Mandarin Stacked**		**Primed**	
	MAXBT	MAXR$'^{R}_{p}$	MAXBT	MAXR$'^{R}_{p}$	MAXBT	MAXR$'^{R}_{p}$
OO	4.5	[14,10,9]	2	[26,18,17,17]	4.37	[32,20,8,8]
SO	4.79	[24,16]	2	[26,24,17]	4.37	[33,20,8,8]
SS	1	[14]	7.70	[27,24,17]	1	[24,8,8]
OS	4.44	[12,9]	7.70	[27,25,17]	4.37	[24,8,8]

Table 5.19: Success of $\langle$MAXBT, MAXR$'^{R}_{p}\rangle$, on stacked and primed RCs under a wh-movement analysis.

Effect Type	**Language**	**Processing Contrast**	**Example #**	$\langle$**MAXT$_{IU}$, AVGR**$\rangle$	
				Promotion	Wh-movement
Primed RCs	English	$SS < OS$	39 < 40	✓	✓
		$OO < SO$	42 < 41	✓	✓
Stacked RCs	English	$SS < OS$	31a < 31b	×	✓
		$OO < SO$	31d < 31c	×	✓
	Mandarin	$SS < OS$	32a < 32b	×	✓
		$OO < SO$	32d < 32c	✓	✓

Table 5.20: Processing preferences for the priming and stacked RCs effects by example, as predicted by $\langle$MAXT$_{IU}$, AVGR$\rangle$.

Priming + Baseline + Stacked A few final considerations can be made by looks at all test cases together.

Based on what discussed above for the promotion analysis, it is easy to deduce that no metric can account for every baseline contrast, together with the stacked RC cases, and the primed RC phenomena. Importantly, none of the reactivation metrics defined in this chapter is able to predict the English $OO < SO$ contrast while also capturing the $OO < SO$ and $SS < OS$ contrasts in Mandarin.

On the other hand, the metric $\langle$MAXT$_{IU}$, AVGR$\rangle$ — which combines the notion of tenure as originally defined in Kobele et al. (2007) with the idea of reactivation introduced in this chapter —

is successful over every contrast, if we adopt a wh-movement analysis of RCs (Table 5.21).

	$\langle$MAXT$_{IU}$, AVGR$\rangle$								
	Primed RCs	**Stacked RCs**		**SRC < ORC**			**ORC < SRC**	**SC/RC < RC/SC**	**Right < Center Embedding**
	English	English	Mandarin	English	Korean	Japanese	Mandarin	English	English
Promotion	✓	×	×	✓	×	×	✓	✓	✓
Wh-movement	✓	✓	✓	✓	✓	✓	✓	✓	✓

Table 5.21: Performance of $\langle$MAXT$_{IU}$, AVGR$\rangle$ for every phenomenon in this chapter.

5.6.5 Additional Tests

Considering how reactivation metrics seem to be unable to account for the whole set of test cases evaluated in this chapter, one thing that we might wonder is whether it makes sense to try and categorize stacked RCs and primed RCs together with processing effects like the difference between right and center embedding. In fact, one could argue that the cognitive mechanisms underlying these processes are radically distinct.

To test this idea, I evaluated the model's performance of every phenomenon involving differences between RCs, while putting aside the sentential complement (SC/RC) and the right/center embedding contrasts. However, the performance of the model is overall the same. Note that this is not actually unexpected, as the previous section already pointed out an issue in trying to account for stacked RCs and primed RCs together.

Importantly, in attempting to incorporate feature reactivation in the MG model of sentence processing, we should not lose sight of questions of psychological plausibility.

In this sense, we might wonder how plausible it is to model reactivation as linear distance effects among similar types of movement dependencies, particularly as human memory dynamics are claimed to be non-linear (Lewis and Vasishth, 2005, a.o.). To test this idea, I then tried to modulate reactivation via a sigmoid function, as one of the most immediate ways to introduce a non-linearity in the MG memory system. However, while sigmoid-based metrics behave somewhat differently from the linear ones, they do not affect the general performance of the model.

As none of these variations significantly changed the overall conclusions of this section, a detailed discussion of these results was omitted.

5.6.6 Interim Summary

A summary of the results in this section is presented in Table 5.22.

		Metrics	Success?	
			Promotion	Wh-movement
Baseline		REACTIVATION, BASE, RANK = 1	×	×
		REACTIVATION, BASE, RANK = 2	×	×
		REACTIVATION, FILTERED, RANK = 2	×	×
		REACTIVATION, FULL, RANK = 2	✓	✓
Stacked	English	REACTIVATION, BASE, RANK = 1	✓	✓
		REACTIVATION, BASE, RANK = 2	✓	✓
		REACTIVATION, FILTERED, RANK =2	✓	✓
		REACTIVATION, FULL, RANK = 2	✓	✓
	Mandarin	REACTIVATION, BASE, RANK = 1	×	×
		REACTIVATION, BASE, RANK = 2	✓	✓
		REACTIVATION, FILTERED, RANK = 2	✓	✓
		REACTIVATION, FULL, RANK = 2	✓	✓
Stacked + Baseline		REACTIVATION, BASE, RANK = 1	×	×
		REACTIVATION, BASE, RANK = 2	×	×
		REACTIVATION, FILTERED, RANK = 2	×	×
		REACTIVATION, FULL, RANK = 2	×	✓
Priming		REACTIVATION, BASE, RANK = 1	✓	×
		REACTIVATION, BASE, RANK = 2	✓	✓
		REACTIVATION, FILTERED, RANK = 2	✓	✓
		REACTIVATION, FULL, RANK = 2	✓	✓
Priming + Stacked		REACTIVATION, BASE, RANK = 1	×	×
		REACTIVATION, BASE, RANK = 2	×	✓
		REACTIVATION, FILTERED, RANK = 2	×	✓
		REACTIVATION, FULL, RANK = 2	×	✓
Priming + Baseline		REACTIVATION, BASE, RANK = 1	×	×
		REACTIVATION, BASE, RANK = 2	×	×
		REACTIVATION, FILTERED, RANK = 2	×	×
		REACTIVATION, FULL, RANK = 2	✓	✓

Table 5.22: Summary of the performance of each cluster of reactivation metrics, over sets of processing phenomena.

As discussed before, reactivation-based metrics did not perform particularly well on the baseline cases. This was not surprising, as none of these processes has been connected to priming-like phenomena in the psycholinguistics literature. Importantly though, there were metrics that did not fail on every case, thus leaving space for possible ranked combinations of original and reactivation metrics. The crucial test cases were, of course, stacked and primed RCs.

Concerning the stacked RC cases, the MG parser enriched with feature reactivation successfully predicts $SS < OS$ and $OO < SO$ for English and Mandarin, on two syntactic analysis of RCs

(promotion and wh-movement).

Crucially, while in the previous section I pointed out how the results of the original MG metrics did not vary significantly depending on the choice of syntactic analysis, that is not the case for reactivation metrics. In particular, while no metric was able to account for the stacked RCs results together with the baseline results when using a promotion analysis of RCs, a few metrics were successful under a wh-movement analysis.

Finally, the extended model was strikingly successful in modeling the primed RCs cases. Importantly, under a wh-movement analysis of RCs, the ranked metric $\langle \text{MAXT}_{IU}, \text{AVGR} \rangle$ was able to predict the correct preferences in the primed RC cases and the baseline cases, as well as in the stacked RC cases. However, this remains an overall unsatisfactory result for several reasons.

First of all, reactivation metrics (and average based reactivation metrics specifically) are very difficulty to interpret with respect to the tree traversal strategy. This undermines one of the main tenants of the MG model: transparency and interpretability.

Moreover, among the numerous new metrics defined in this chapter, only a few are actually able to account for the data under examination. While the overall goal of the approach is to find a small number of metrics that account for the majority of cross-linguistic asymmetries, the fact that these results heavily depend on a very specific analysis requires deeper investigation.

Finally, MAXT_{IU} — that is, highest tenure value among those of internal and unpronounced nodes — *might* be a reasonable memory metric when thinking about offline complexity. However, it is unclear whether it is cognitively realistic to think of the memory burden associated to internal nodes as the main driving force behind complexity profiles in sentence processing. Thus, further thought needs to be given to the psychological plausibility of these metrics.

In sum, while it can be claimed that reactivation-based metrics were successful with respect to the initial goals of the chapter, it seems that the extended MG model still comes short as a comprehensive model of offline processing. Because of this, it is interesting to explore alternative ways of modulating memory burden. This is the focus of the next section.

5.7 A Different Approach: Weighted Metrics

In the previous section, the idea of reactivation metrics came from noticing a shortcoming in the existing implementation of the MG model: namely, ignoring the effect of features on memory load. In a similar fashion, this section explores changes to another aspect of the original metric system: ranked evaluation.

5.7.1 Weighted Metrics: Principles

Ranked metrics were first introduced in the MG parser by Graf et al. (2015b), and resemble constraint ranking in Optimality Theory (OT; Prince and Smolensky, 2008) — a lower ranked metric matters only if all higher ranked metrics have failed to pick out a unique winner. Introducing a ranking system leads to a significant expansion of the metric space. However, Graf et al. (2017) suggest that a few, selected metrics of small rank are in fact sufficient to account for a variety of processing phenomena.

When it comes to memory usage though, it is reasonable to wonder what is the cognitive plausibility of a strict ranking system. Importantly, complexity metrics are *not* constraints, at least not in the way the latter are usually understood in linguistic theory. Moreover, while strict constraint ranking has been enormously successful in linguistics, recent work on constraint-based formalisms has seen the growth of different kinds of approaches, which might be more suitable to account for linguistic data.

In particular, here I am interested in the ideas behind *Harmonic Grammar* (Legendre et al., 1990; Pater, 2008, 2009, 2016; Potts et al., 2010). This framework abandons OT strict ranking approach, and instead assigns each constraint a numerical weight reflecting its relative strength (see also Guy, 1997, a.o.).

Consider two derivations X and Y, and two metrics M_1 and M_2 such that: $M_1(X) = 1, M_2(X) = 1$, $M_2(Y) = 1$, and $M_2(Y) = 1$. As seen before, if we consider the ranked metric $<M_1, M_2>$, the MG parser will evaluate each derivation on M_1 first. In doing so, it will decide that the candidate derivation with the lowest value for that metric is the winner. In OT terms, this is like assigning the losing candidate a violation. Since in the example above $M_1(X)$ is lower than $M_1(Y)$, a parser

equipped with this ranked metric predicts $X < Y$, and the values of M_2 end up not mattering at all.

	M_1	M_2
X		*
Y	*	

Assume now that, instead of being ranked, the two metrics are assigned weights such that $weight(M_1) = 1$, and $weight(M_2) = 2$. This time, derivations are evaluated independently over each metric, and are assigned a violation relative to the metric they fail upon. Then, violation counts are multiplied by the corresponding weights, and add up the total across metrics. This yields a kind of penalty score, which in constraint-based approached is usually labelled harmony. The winning candidate is the least penalized one; i.e. the one with the lowest harmony.

In the example above, Y is assigned a violation on M_1 (as $M_1(Y) > M_1(X)$), but X is assigned a violation on M_2 (as $M_2(X) > M_2(Y)$). M_2 violations are penalized more though, since M_2 has double the weight of M_1, and the system ends up predicting $Y < X$:

	M_1	M_2	**Harmony**
	$\times 1$	$\times 2$	
X		*	$1 \times 2 = 2$
Y	*		$1 \times 1 = 1$

This section follows these ideas, in proposing combinations of weighted complexity metrics. For a cognitive perspective, one could think of this approach as an attempt to find a derivation with optimal memory consumption — with distinct metrics formalizing different types of memory bounds on the underlying sentence processing mechanisms. As a first evaluation step, the system is tested on the sets of phenomena discussed so far in the chapter.

5.7.2 Weighted Metrics: Model Evaluation

This section evaluates the MG parser equipped with a set of weighted metrics, over the processing phenomena discussed previously in the chapter: baseline cases, stacked RCs, and primed RCs. As before, there are several modeling choices that need to be made.

First of all, as we are working with a weighted system, it might be conceivable to consider complex metrics spanning *every* possible base metric defined so far. However, once we start putting weights into the equation, the number of conceivable metrics of that kind quickly becomes gigantic. Thus, such an approach would be both uninterpretable, and computationally unrealistic. Instead, I will only consider metrics of rank 2.

To distinguish them from the ranked versions, I will refer to such metrics as $(M_1, M_2)_{w1,w2}$ — where the subscripts $w1$ and $w2$ are the weights assigned to the first and second metric in the tuple, respectively. To clarify, the order of the metrics in the tuple has no significance in the decision process, and it is just a useful shorthand to relate each metric to its weight.

Moreover, for succinctness, in this section I will only the results of two specific choices of weights: a case in which both metrics have the same weight — $(M_1, M_2)_{1,1}$ — and a case in which the second metric is weighted double — $(M_1, M_2)_{1,2}$. As base metrics, I will consider the set of original metrics, and the set of reactivation-based metrics as defined in previous sections.

Finally, RC constructions will be built according to a promotion analysis and a wh-movement analysis. The feature annotation for each derivation is the same as used in Section 5.6.1.

Consistently with what was done in previous sections, I assign labels to clusters of metrics as follows:[7]

- ORIGINAL, $(M_1, M_2)_{w1,w2}$: just original metrics (as in Section 5.4), filtered and unfiltered. $w1, w2$ specify the weights assigned to each metric;
- REACTIVATION, BASE,$(M_1, M_2)_{w1,w2}$: just reactivation metrics (as in Section 5.5.1), filtered and unfiltered;
- REACTIVATION, FULL, $(M_1, M_2)_{w1,w2}$: original metrics and reactivation metrics, filtered and unfiltered.

[7]Due to the number of possible metrics generated by the weighted approach, in this section I discuss results just based on each cluster's performance. However, these results are available, together with the new implementation of the mgproc package, at https://github.com/aniellodesanto/mgproc_weighted.

5.7.2.1 Modeling Results: Original Metrics

As a starting point, we want to evaluate the performance of the original set of complexity metrics, with this new approach to encoding metric interactions.

$(\mathbf{M_1}, \mathbf{M_2})_{\mathbf{1,1}}$ Metrics First, it is reasonable to explore an unranked system in which each metric is assigned the same weight. This is the most immediate transition from the previous system: strict ranking is discarded in favor of evaluating the contribution of each individual metric towards deciding in favor of a process over another.

Unsurprisingly, this set of unranked metrics does not perform well on the contrasts at hand. In particular, while there are metrics that can account for some of the baseline processes individually[8], no metric is able to account for all of them at the same time. Moreover, no weighted metric makes the correct predictions for the stacked RC cases (independently of the language), and for the primed RC cases — neither separately nor together. These results are consistent across syntactic analysis of RCs.

$(\mathbf{M_1}, \mathbf{M_2})_{\mathbf{1,2}}$ Metrics Things change when we differentiate the weight assigned to each metric. In particular, in this chapter I only look at weighted combinations in which the weight of the second metric in the tuple is double that of the first metric. Even so, there is a variety of weighted combinations of the original metrics that produces the right results for each set of processes (baseline, stacked RCs, and primed RCs) individually *and* all together.

There are a few considerations to be made here. First of all, one might wonder whether weighting the second metric more than the first is artificially reproducing the effects of the strict ranking approach — for instance in having the heavier metric resolve ties missed by the lighter one. However, looking at the performance of these metrics over individual processes seems to disprove this idea. For instance, there is no effect of $(\text{MAXT}, \text{SUMS})_{1,2}$, while the ranked version of this metric was argued to be incredibly successful in previous chapters.

Moreover, among all successful combinations, none of the individual metrics is based on size or payload. In fact, every weighted metric that correctly accounts for the processing phenomena

[8] Note that this is also unsurprising, as some of the baseline contrasts are also captured by individual, rank 1 metrics.

considered here, is a combination of two tenure-based metrics. A reasonable question is then whether — from a cognitive perspective — it makes sense to be balancing metrics based on the same notion of memory usage.

Finally, a variety of weighted combinations of the original metrics can account for the whole set of processing phenomena under either the promotion analysis (e.g., (MAXT, AVGT)$_{1,1}$), or the wh-movement analysis (e.g., (AVGT, SUMT)$_{1,1}$). However, there seems to be no metric that gives the correct results independently of syntactic analysis. That is, there is no intersection between $1,1$ metrics that work under a promotion analysis, and those that work under a wh-movement analysis.

5.7.2.2 Modeling Results: Reactivation Metrics

Given the results above, it is worth exploring the contribution of reactivation metrics by themselves.

$(\mathbf{M_1},\mathbf{M_2})_{\mathbf{1,1}}$ Metrics As for the original ones, combinations of equally weighted reactivation metrics cannot account for the whole set of results under consideration. Again, this does not imply that every metric fails on every contrast. However, there is a general inability of $1,1$ metrics to account for clusters of effects.

$(\mathbf{M_1},\mathbf{M_2})_{\mathbf{1,2}}$ Metrics When we consider $1,2$ weights, weighted reactivation metrics behave strikingly similar to the original metrics — at least in terms of general coverage, if not of 1-to-1 performance of each contrast.

Here, the majority of the successful metrics seem to belong to a combination of pure REACTIVATION (R) measures. As for the tenure-based combinations above, a question remains about the plausibility of combined metrics mixing different estimates (max, sum, avg) for the same type of memory measure. Importantly, the performance of each weighted metric still varies significantly based on the syntactic analysis of choice, and no combined metric is able to correctly account for all results across both approaches.

5.7.2.3 Modeling Results: Original and Reactivation Metrics

Finally, it is worth investigating the effect of weighted combinations of original and reactivation metrics. As before, no metric in the $(\mathbf{M_1}, \mathbf{M_2})_{\mathbf{1,1}}$ set is able to predict all the processing contrasts correctly. This comes to no surprise, given the results of each set of $1,1$ metrics evaluated above.

$(\mathbf{M_1}, \mathbf{M_2})_{\mathbf{1,2}}$ Metrics Modulating the weights over the second metric in the tuple is once again effective. In fact, there are numerous similarities between the behavior of these mixed metrics, with that of the ones explored above.

Importantly though, mixed $(M_1, M_2)_{1,2}$ metrics give consistent results over every processing contrast considered, across both the promotion and the wh-movement analysis of RCs. Interestingly, size-based and boost-based metrics do not have any useful effect. All of the weighted metrics which succeed independently of RC analysis are a combination of TENURE and REACTIVATION measures (e.g., (MAXT′, MAXR′)$_{1,2}$).

5.7.3 Interim Summary

Table 5.23 summarizes the performance of clusters of weighted metrics on the processing phenomena studied in this chapter.

There is a striking symmetry across types of weighted metrics. In particular, no $(M_1, M_2)_{1,1}$ combination — with identical weights for both metrics — was able to produce reliable result across every processing phenomena under consideration. While not especially surprising, this result is still informative, as it suggests that it is not enough to simply discard a strict ranking approach and evaluate each individual metric at the same time.

Thus, this section explored unbalanced weight assignment in the form of $(M_1, M_2)_{1,2}$ metrics, leading to surprisingly successful results. Combinations of metrics of the same type (original vs. reactivation based) produced overall interesting results, each succeeding on all contrasts under different analyses of RCs. However, the most encouraging results came with weighted combinations mixing original and reactivation metrics. Metrics in this set not only were able to account for every single contrast, but were able to provide sound predictions across syntactic

choices.

Importantly, and differently from what we obtained for reactivation metrics in a strict ranking system, success in this system was also not restricted to a single metric. In fact, there is a variety of weighted metrics successful on the processing contrasts. However, the risk of empirical indeterminacy seems to be contained anyway, as these metrics are all based on two specific notions of memory usage and memory reactivation (T and R). Thus, future work could focus on exploring which of the several available measures based on these notions (Max, Sum, Avg) is more in line with psychologically plausible theories of sentence processing.

	Metrics	**Success?**	
		Promotion	Wh-movement
Baseline	ORIGINAL, $(M_1,M_2)_{1,1}$	×	×
	ORIGINAL, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, BASE, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, BASE, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, FULL, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, FULL, $(M_1,M_2)_{1,2}$	✓	✓
Stacked	ORIGINAL, $(M_1,M_2)_{1,1}$	×	×
	ORIGINAL, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, BASE, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, BASE, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, FULL, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, FULL, $(M_1,M_2)_{1,2}$	✓	✓
Priming	ORIGINAL, $(M_1,M_2)_{1,1}$	×	×
	ORIGINAL, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, BASE, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, BASE, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, FULL, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, FULL, $(M_1,M_2)_{1,2}$	✓	✓
Priming + Stacked + Baseline	ORIGINAL, $(M_1,M_2)_{1,1}$	×	×
	ORIGINAL, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, BASE, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, BASE, $(M_1,M_2)_{1,2}$	✓	✓
	REACTIVATION, FULL, $(M_1,M_2)_{1,1}$	×	×
	REACTIVATION, FULL, $(M_1,M_2)_{1,2}$	✓	✓

Table 5.23: Summary of the performance of each cluster of weighted metrics, over sets of processing phenomena.

5.8 Discussion

This chapter discussed a set of processing phenomena that present a challenge to the current implementation of the MG model: stacked RC constructions in English and Mandarin Chinese, and priming of subject and object RCs in English.

These effects are a good litmus test for the performance of the MG parser, as their processing profiles seem to be related to the interaction of similar structural configurations during parsing — a process that the MG model should be currently unable to encode. RCs were chosen as test cases, as at their core they involve constructions the MG model has been extensively tested upon.

First, I showed how the current set of complexity metrics (labeled above *original* metrics) is unable to correctly account for the new set of phenomena Table 5.8. Interestingly, it turned out that some original metrics (e.g., AVGT) are indeed able to predict a few of the correct contrasts correctly. However, even varying the way RCs were built, no metrics was able to account for the priming cases and the stacked RCs cases together with the processing phenomena chosen as baseline. Moreover, these results highlighted fundamental differences in how the metric performs for the stacked cases in English and Mandarin Chinese, which will need to be carefully explored in the future.

Building on the unsuccessful performance of the existing MG metrics, the chapter explored different ways to extend the MG model, by rethinking the way existing complexity metrics encode memory usage.

First, I presented metrics encoding facilitatory effects due to the repetition of identical movement features. These *reactivation* metrics are based on insights on memory reactivation coming from the psycholinguistic literature on structural priming (Troyer et al., 2011; Reitter et al., 2011), and they were expected to easily account for the structural parallelism in stacked and primed RCs. However, the performance of these metrics was overall less encouraging than originally anticipated (Table 5.22.)

As expected, reactivation-based metrics alone did not perform particularly well on the baseline cases. Moreover, they were not able to account for the plethora of phenomena under analysis, when considered all at the same time.

Things got better when considering a combination of original and reactivation metrics. Ranked combinations from these sets were able to account for the stacked RC and the priming contrasts when considered in isolation, but highlighted a big divide in terms of which metrics were able to predict stacked effects, and which ones were predicting priming effects. This result is particularly interesting, as primed RCs and stacked RCs are structurally very similar to each other. In the future, it might be worth to explore more in detail the syntactic assumptions behind these two constructions — and, possibly, to design sentence processing studies aimed at uncovering similarities and differences between the two.

Significant differences in performance also emerged when the same phenomenon was evaluated using a promotion analysis, or a wh-movement analysis of RCs. In fact, only when adopting a wh-movement analysis it was possible to find a metric with correct predictions for every single contrast under consideration.

Obviously, there are many conceivable variants of the metrics defined in this chapter, as well as alternative definitions to the core concepts of reactivation and boost. Crucially, the metrics defined here restrict reactivation to nodes associated to movement features. This is in line with the way movement features are kept track of in Kobele et al. (2007)'s formalization of the MG parser as a bottom-up tree transducer. Thus, going back to mathematics behind automata-based characterizations of MGs processing mechanisms could give us concrete ways to explore these ideas, while also strengthening the link between mathematical and cognitive approaches to the study of memory systems. Importantly though, among the many possible variants of the metrics above, in the future it would be interesting to generalize the definition of reactivation to Merge features.

Finally, the last section of the chapter explored the idea of modeling interactions between metrics not as a ranked system, but by introducing weighted evaluations.

In this perspective, the most encouraging results came from weighted combinations of original and reactivation metrics — in which one of the two metrics was weighted as double the other (Table 5.23). By relying on reactivation (R) and tenure T, metrics in this set not only were able to account for every single contrast under consideration, but could also perform consistently across syntactic choices.

While the fact that the simple instances of weight assignment tested above were successful on a variety of constructions is inspiring, this chapter's attempt to define a weighted system balancing interacting memory metrics is clearly still preliminary. As mentioned, it is possible to conceive of many alternative ways to assign weights to unranked metrics.

Moreover, in the constraint-based approaches to grammatical knowledge that inspired the system used here, weights are usually assumed to be learned. Importantly though, the metrics used by the MG parser are meant to encode cognitive limitations on the human sentence processing system. Thus, it is unclear whether it would make sense to assume that weights for such metrics have to be learned, instead of simply reflecting general architectural bounds.

In this sense, the idea of parsing being guided by a mechanism striving for harmony over sets of cognitive constraints is in line with existing work on parsers operating in a continuous representational space (Hale and Smolensky, 2001; Gerth and Beim Graben, 2009; Tabor, 2009; Cho et al., 2017, 2018). A more extensive study of the parallels between the MG model and such connectionist-like approaches could lead to valuable insights into the role of constraint-based theories in psycholinguistics, and on the psychological plausibility of weighted combinations of memory bounds.

Lastly, one possible objection to the case studies presented above is the narrow focus on constructions involving relative clauses. In this sense, it was mentioned before that the most studied cases of structural priming involve constructions like prepositional-dative vs double-object alternations, active-passive alternations, or garden-path sentences (Pickering and Ferreira, 2008). Recall though that feature reactivation as defined here primarily refers to movement features, and it is not immediately obvious that the processing asymmetries across these more studied phenomena involve movement in a significant way. These constructions also involve argument alternations that introduce confounds between surface syntax with thematic mappings (Ziegler et al., 2017; Ziegler and Snedeker, 2018; Oltra-Massuet et al., 2017). Thus, the choice of ignoring these cases is consistent with the general tenant of this dissertation (and of the MG processing model so far): that is, for the time being, we want to set aside processing factors that are not purely structural.

Moreover, priming has been used as a technique to probe the nature of syntactic representations, under the assumption that "if processing one stimulus affects the subsequent processing of another

stimulus, then these stimuli share some aspect of their representation" (Branigan and Pickering, 2017). However, the phenomena just mentioned show us how such effects are complex, and we are far from understanding whether they truly tap into the representational details syntactictians care about (Mahowald et al., 2016; Ziegler et al., 2017). From this perspective, enriching the MG models with metrics sensitive to Merge features will be a fundamental step in trying to predict facilitatory phenomena due to structural priming more in general, and could contribute unexpected insights into the mechanisms driving such effects.

Chapter 6

Conclusions and Future Work

An important problem at the intersection between theoretical linguistics and psycholinguistics is whether the fine-grained grammatical analyses posited by syntactitians have any relevance to the study of the cognitive processes underlying language processing. In fact, in forms more or less explicit, this question has always been at the core of generative linguistics.

> If the study of actual linguistic behavior is to proceed very far, it must clearly pay more than passing notice to the competence and knowledge of the performing organism. We have suggested that a generative grammar can give a useful and informative characterization of the competence of the speaker-hearer, one that captures many significant and deep-seated aspects of his knowledge of his own language. [...] The question is, therefore, how does he put his knowledge to use in producing a desired sentence or in perceiving and interpreting the structure of presented utterances? How can we construct a model for the language user that incorporates a generative grammar as a fundamental component?
>
> (Miller and Chomsky, 1963, pg. 464-465)

In this sense, Miller and Chomsky's Derivational Theory of Complexity (DTC) was an attempt to provide *"a model for the language user that incorporates a generative grammar as a fundamental component"*, by formulating a transparent mapping between grammatical operations and cognitive processes. While the DTC was dismissed on seemingly empirical grounds, it is clear that its shortcomings were due to the lack of an operationalized theory of how syntactic transformations

affect cognitive cost, rather than to the hypothesis of a strong connection between grammatical and processing complexity.

This dissertation recasts such questions in a modern framework, by expanding on an existing computational model in order to specify a transparent linking hypothesis between grammatical structure and processing complexity. In particular, I explored a model which adopts a fully formalized theory of grammatical structures (MGs), an algorithm detailing how such structures are built over time (a top-down parser), and an explicit theory of how structure-building operations affect cognitive load — as a vast set of complexity metrics measuring memory usage. The striking success of this model in predicting a variety of complexity effects in sentence processing shows the real potential of the DTC's assumptions, once each component of the theory is explicitly laid down.

6.1 The Road So Far

The MG approach grounds its results in a specific theory of syntactic representations (MG derivation trees) *and* a psychologically plausible theory of cognitive cost (memory burden). In doing so, it tries to address some of the dangers that come from adopting computational models to study complex cognitive processes — as, for instance, empirical indeterminacy due to the opaqueness and arbitrariness of the modeling choices.

> [...] this is a confusion of two quite separate issues, simulation and explanation. As scientists, we are not merely interested in simulating human behavior — in constructing a black box that behaves exactly as people behave, has the same profiles of complexity and so forth. What we are rally interested in as scientists is explanation — in developing models that help us understand how it is that people behave that way, not merely demonstrating that we can build an artifact that behaves similarly. We don't want to replace one black box, namely a person, by another black box, namely the artifact that we have built. We should look for modular theories that account for the observed interactions in terms of the interleaving of information from separate, scientifically comprehensible systems.
>
> (Kaplan, 1995, pg. 348)

Encouragingly, previous literature had shown that this framework is successful in modeling several processing phenomena cross-linguistically; spanning from the difference between center-embedding and right-embedding constructions (Kobele et al., 2013), to attachment ambiguity (Lee, 2018), and quantifier scope resolution (Pasternak and Graf, 2020).

As mentioned, the current work was motivated by the desire to explore the extent to which this specific model can provide a transparent, empirically valid reframing of past theories trying to connect grammatical operations to cognitive difficulty as directly as possible (Miller and Chomsky, 1963).

First, by looking at a variety of processing asymmetries in Italian, I showed how the model's sensitivity to fine-grained grammatical assumptions (e.g., the precise syntactic analyses of Italian postverbal subjects) are crucial in guiding the MG parser towards correct processing predictions. This also allowed me to show how some processing phenomena that are usually attributed to ambiguity resolution strategies can be accounted for through a purely deterministic process relying on memory load — thus opening questions about whether it would be possible to use this approach to characterize phenomena where ambiguity really is the decisive factor.

Secondly, I argued that this bridge between syntactic theory and psycholinguistic insights works in both directions, by showing how we can bring experimental data to address questions about the nature of grammatical representations. In particular, I addressed the debate about whether the source of gradience in human acceptability judgments is situated in the grammar itself, by showing that gradient effects reported for the acceptability of Island constructions in English can be derived from memory-load factors as measured by the MG parser. As mentioned in Chapter 4, these results then open new opportunities to compare different cognitive hypotheses about grammatical and extra-grammatical constraints on sentence acceptability (Boston, 2012).

Importantly, this work also leads to crucial questions about the status of grammatical and ungrammatical structures in our parsing models. In particular, I made the assumption that even ungrammatical structures (e.g., sentences violating an island constraint) can be correctly analyzed by the parser. While there is evidence that humans assign an interpretation even to this kind of sentences (hence, from a generative perspective, they should have some structure assigned to them), the question of what role they occupy in the parser's hypothesis space is not a trivial one.

I suggested that structures that violate specific grammatical constraints reduce acceptability in a way that is not determined by processing factors. One possible, although somewhat arbitrary hypothesis, is that each grammatical violation is assigned a specific cost. This obviously opens new questions about the origins of such costs, about cost differences across violation types, etc. Alternatively, an interesting future line of inquiry would be to investigate whether ungrammatical structures could be costly because of different computational bounds on the cognitive architecture. For instance, Graf and De Santo (2019) argue that syntactic constraints — for instance, the Specifier Island Constraint — increase the efficiency of the parser by making sure that licensed grammatical structures only contain syntactic dependencies that can be recognized by a sensing tree automaton. This line of thought might then allow us to give a deeper look into the relation between expressivity of the grammar, syntactic constraints, and parsing efficiency (cf. Tabor, 2009).

Finally, building on these successes, I asked how extended the variety of processing phenomena covered by the model can be. Starting from results on stacked relative clauses and priming effects, I argued that the plausibility of the MG parser as a good cognitive model of sentence processing is actually severely reduced, due to the current way the model estimates processing complexity. I then addressed such limitations, by proposing new complexity metrics encoding how grammatical features affect overall memory usage.

Interestingly, syntactic priming has been linked to gradience effects in the acceptability of island constructions in terms of *satiation effects* (Luka and Barsalou, 2005; Do and Kaiser, 2017). Independently of priming, the feature composition of lexical items has also been claimed to influence acceptability judgments, and processing complexity more in general (Rizzi, 1990; Friedmann et al., 2009). Given that the MG model already proved informative in modeling gradient effects, this would be an interesting area to explore.

Moreover, as suggested in the final sections of Chapter 5, rethinking the way the MG approach handles the interaction of different factors contributing to memory load — specifically, moving from a ranked system to a weighted one — might allow this parsing model to integrate insights from connectionist-like approaches to structural processing (Villata et al., 2018, a.o.).

Additionally, the fact that priming phenomena inspired a re-organization of the model's memory

mechanisms confirms the importance of extending the empirical coverage of the approach. In this sense, voicing-mismatches effects in the processing of ellipsis-constructions have been gaining attention in the sentence processing literature (see Poppels and Kehler, 2019, and references therein), and would prove to be a fascinating challenge for the MG parser. These effects might be harder to model, since they require strong assumptions about the behavior of the parser at the ellipsis site. But success of the parser on these phenomena would provide insights that could hardly come from standard experimental approaches.

Crucially, while in this dissertation I followed existing work on MG parsing and adopted a top-down parser, it has been argued that left-corner parsing is closer to the way humans process sentences. In this sense, it is important to note that what we know about the human parser just suggests a combination of top-down (predictive) and bottom-up (eager lexical-integration) strategies (Resnik, 1992). A left-corner parsing strategy is one possible way of combining these.

Thus, this work's focus on exploring how far the top-down component of the human parser can go is valuable by itself, as in contributes to our understanding of the role of the predictive component of the parser during processing. However, it would also be interesting to evaluate recent results on left-corner parsing for MGs through the lens of processing models (Hunter, 2018a; Stanojević and Stabler, 2018; Hunter et al., 2019).

Finally, according to the research tradition this dissertation builds on, part of the appeal of having a mathematically worked out model is that it should make it possible to distinguish between competing syntactic proposals based on their psycholinguistic predictions (Bresnan, 1978; Berwick and Weinberg, 1983; Rambow and Joshi, 1994; Kobele et al., 2013; Graf et al., 2017).

> But the grammatical realization problem can clarify and delimit the grammatical characterization problem. We can narrow the class of possible theoretical solutions by subjecting them to experimental psychological investigation as well as to linguistic investigation.
>
> (Bresnan, 1978, pg. 59)

As outlined by Kobele et al. (2013), the MG model could contribute to these questions by clarifying which aspects of sentence structure — as hypothesized by different grammatical theories — correctly modulate processing difficulty. Encouragingly, there have been some initial attempts

to leverage the MG model's interpretability to compare the predictions of alternative syntactic approaches (De Santo and Shafiei, 2019), which thus lay the ground for extensive future work using experimental results to directly inform theoretical stances.

6.2 Looking Ahead

The results in the previous chapters open up numerous avenues for future research. Therefore, it is important to ask what the natural next step for this enterprise should be.

In doing this, it is worth remarking once more that a computational model is *not* a theory on its own — and it should not be confused with one. However, transparent computational models can help (or even drive) theory building, by forcing us to commit to explicit formulations of our theoretical assumptions (van Rooij and Baggio, 2020; Guest and Martin, 2020). Computational modeling is thus a powerful, necessary tool in exploring the intricacies of human cognition. In looking at possible venues of future research then, priority should be given to refining those aspect of the model that will put it in a position to directly contribute to essential discussions at the intersection between syntactic theory and psycholinguistic inquiry. In this sense, given the degrees of freedom available to the model, it is important to address concerns about the cognitive plausibility of the variety of idealizations the current implementation of the MG parser comes with.

As pointed out before, the MG parser is assumed to be equipped with a perfect oracle, and thus cannot directly account for ambiguity in sentence comprehension. While this dissertation argued for the advantage of focusing on deterministic parses, in pursuit of a fully comprehensive model of human cognition it will obviously be important to see how a non-deterministic version of this approach — possibly using probabilistic weights learned from a corpus (Torr, 2017, 2018; Torr et al., 2019) — would perform on processing phenomena usually associated with structural ambiguity. This will also encourage a more straightforward discussion between the MG approach, and probabilistic models of sentence complexity.

Adding probabilities into the model would allow us to compare memory-based complexity metrics to measures of cognitive load like *entropy* and *surprisal*, adapted from expectation-based theories of processing difficulty (Hale, 2016, a.o.). In this sense, I mentioned before how Gerth

(2015) claims that a combination of tenure and surprisal over MG derivation trees could be used as reliable predictors for off-line (overall across a sentence) complexity and on-line (i.e., word-by-word) complexity. On-line metrics would also give us access to a greater variety of experimental data, as they can be used to index measures of processing load like word-by-word reading times latencies, ERP amplitudes, and fMRI/MEG activation levels (Brennan et al., 2016, a.o.). Thanks to such extensions, the MG model would then be able to address the connection between theoretical syntax and experimental linguistics more generally.

Finally, Chapter 5 highlighted the variety of ways in which the current memory metrics can be extended to take into account richer grammatical information. In order to test the model on a growing set of processing contrasts — and develop it so that it can contribute to a variety of theoretical debates in the literature — accounting for information about the feature component of a derivation is going to be essential. In this sense, developing and testing metrics that are sensitive to Merge features is *the* fundamental next step in this enterprise, and it would put us one step forward towards an explicit formulation of a complete theory of syntactic representations.

Circling back to the discussion set up in Chapter 1 and Chapter 2, theoretical, experimental, and computational approaches are often seen as disjoint in linguistic inquiry — if not incomparable and/or in competition with each other. Instead, by significantly extending our understanding of the mechanisms driving the MG model — and thus creating exciting new opportunities for future research on the relation between grammar and processing behavior — the results in this dissertation stress how inter-whined these different perspectives can (and should) be.

> Within the program of research proposed here, joint work by linguists, computer scientists, and psychologists could lead to a deeper scientific understanding of the role of language in cognition.
>
> (Bresnan, 1978, pg. 59)

In this sense, the past decades brought extraordinary refinements of our linguistic theories, a deeper understanding of the formal properties of syntactic representations and general psycholinguistic mechanisms, and great advances in computational and experimental methodologies. Thus, we are now in a position to truly embrace Bresnan (1978)'s suggestions, and confidently pursue a multidisciplinary path towards the cognitive study of language.

Bibliography

David Adger. 2003. *Core syntax: A minimalist approach*, volume 33. Oxford: Oxford University Press.

David Adger and Peter Svenonius. 2001. Features in minimalist syntax. In *The Oxford Handbook of Linguistic Minimalism*, 1, pages 27–51.

Alfred V Aho and Jeffrey D Ullman. 1973. *The theory of parsing, translation, and compiling*. Prentice-Hall.

Theodora Alexopoulou and Frank Keller. 2003. Linguistic complexity, locality and resumption. In *Proceedings of WCCFL*, volume 22.

Theodora Alexopoulou and Frank Keller. 2007. Locality, cyclicity, and resumption: At the interface between the grammar and the human sentence processor. *Language*, pages 110–160.

John Robert Anderson. 1996. *The architecture of cognition*, volume 5. Psychology Press.

Lars-Gunnar Andersson, Elisabet Engdahl, and Eva Ejerhed. 1982. Readings on unbounded dependencies in scandinavian languages.

Francesco Antinucci and Guglielmo Cinque. 1977. Sull'ordine delle parole in italiano: l'emarginazione. *Studi di grammatica italiana VI, pp. 121-146*.

Fabrizio Arosio, Flavia Adani, and Maria Teresa Guasti. 2009. Grammatical features in the comprehension of Italian Relative Clauses by children. *Merging Features: Computation, Interpretation, and Acquisition*, pages 138–158.

Fabrizio Arosio, Francesca Panzeri, Bruna Molteni, Santina Magazù, and Maria Teresa Guasti. 2017. The comprehension of Italian relative clauses in poor readers and in children with specific language impairment. *Glossa: a journal of general linguistics*, 2(1).

Emmon Bach, Colin Brown, and William Marslen-Wilson. 1986. Crossed and nested dependencies in german and dutch: A psycholinguistic study. *Language and Cognitive Processes*, 1(4):249–262.

Elizabeth Bates and Brian MacWhinney. 1987. Competition, variation, and language leaning. *Mechanisms of language acquisition*.

Adriana Belletti. 1988. The case of unaccusatives. *Linguistic inquiry*, 19(1):1–34.

Adriana Belletti and Carla Contemori. 2009. Intervention and attraction. on the production of subject and object relatives by Italian (young) children and adults. In *Language acquisition and development, 3. Proceedings of Gala*, pages 39–52.

Adriana Belletti and Chiara Leonini. 2004. Subject inversion in L2 Italian. *EUROSLA yearbook*, 4:95–118.

Robert C. Berwick and Amy S. Weinberg. 1982. Parsing efficiency, computational complexity, and the evaluation of grammatical theories. *Linguistic Inquiry*, 13:165–291.

Robert C. Berwick and Amy S. Weinberg. 1983. The role of grammar in models of language use. *Cognition*, 13:1–61.

Robert C. Berwick and Amy S. Weinberg. 1985. The psychological relevance of transformational grammar: a reply to Stabler. *Cognition*, 19:193?204.

Kathryn Bock, Gary S Dell, Franklin Chang, and Kristine H Onishi. 2007. Persistent structural priming from language comprehension to language production. *Cognition*, 104(3):437–458.

Kathryn Bock and Zenzi M Griffin. 2000. The persistence of structural priming: Transient activation or implicit learning? *Journal of experimental psychology: General*, 129(2):177.

Cedric Boeckx. 2012. *Syntactic islands*. Cambridge University Press.

Marisa Ferrara Boston. 2010. The role of memory in superiority violation gradience. In *Proceedings of the 2010 Workshop on Cognitive Modeling and Computational Linguistics*, pages 36–44. Association for Computational Linguistics.

Marisa Ferrara Boston. 2012. *A computational model of cognitive constraints in syntactic locality*. Ph.D. thesis, Cornell University.

Silke Brandt, Sanjo Nitschke, and Evan Kidd. 2017. Priming the comprehension of german object relative clauses. *Language Learning and Development*, 13(3):241–261.

Holly P Branigan and Martin J Pickering. 2017. An experimental approach to linguistic representation. *Behavioral and Brain Sciences*, 40.

Jonathan R Brennan, Edward P Stabler, Sarah E Van Wagenen, Wen-Ming Luh, and John T Hale. 2016. Abstract linguistic structure correlates with temporal activity during naturalistic comprehension. *Brain and Language*, 157:81–94.

J. Bresnan. 1982. *The Mental representation of grammatical relations*. MIT Press series on cognitive theory and mental representation. MIT Press.

Joan Bresnan. 1978. A realistic transformational grammar. *Linguistic theory and psychological reality*, pages 1–59.

Donald Eric Broadbent. 2013. *Perception and communication*. Elsevier.

Sarah M Callahan, Lewis P Shapiro, and Tracy Love. 2010. Parallelism effects and verb activation: The sustained reactivation hypothesis. *Journal of psycholinguistic research*, 39(2):101–118.

David Caplan and Gloria S Waters. 1999. Verbal working memory and sentence comprehension. *Behavioral and brain Sciences*, 22(1):77–94.

Anna Cardinaletti. 1998. A second thought on" emarginazione": Destressing vs." right dislocation". *Working Papers in Linguistics, 8.2, 1998, pp. 1-28*.

Craig G Chambers, Michael K Tanenhaus, and James S Magnuson. 2004. Actions and affordances in syntactic ambiguity resolution. *Journal of experimental psychology: Learning, memory, and cognition*, 30(3):687.

Rui P Chaves and Jeruen E Dery. 2019. Frequency effects in subject islands. *Journal of Linguistics*, 55(3):475–521.

Pyeong Whan Cho, Matthew Goldrick, Richard L. Lewis, and Paul Smolensky. 2018. Dynamic encoding of structural uncertainty in gradient symbols. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 19–28, Salt Lake City, Utah. Association for Computational Linguistics.

Pyeong Whan Cho, Matthew Goldrick, and Paul Smolensky. 2017. Incremental parsing in a continuous dynamical system: sentence processing in gradient symbolic computation. *Linguistics Vanguard*, (1).

Noam Chomsky. 1956a. Three models for the description of language. *IRE Transactions on Information Theory*, 2:113–124.

Noam Chomsky. 1956b. Three models for the description of language. *IRE Transactions on information theory*, 2(3):113–124.

Noam Chomsky. 1965. *Aspects of the Theory of Syntax*, volume 11. MIT Press.

Noam Chomsky. 1975. *The logical structure of linguistic theory*. University of Chicago Press.

Noam Chomsky. 1977. On wh-movement. *Formal syntax*, pages 71–132.

Noam Chomsky. 1986. Barriers. *Cambridge, Mass./London, England*.

Noam Chomsky. 1993. *Lectures on government and binding: The Pisa lectures*. 9. Walter de Gruyter.

Noam Chomsky. 1995. *The Minimalist Program*. Cambridge, MA: MIT Press.

Noam Chomsky, Stephen Anderson, and Paul Kiparsky. 1973. A festschrift for morris halle.

C. Clifton, L. Frazier, and K. Rayner. 1994. *Perspectives on sentence processing*. L. Erlbaum Associates.

Charles Jr. Clifton. 2015. Sentence comprehension. In *Psychology of. International Encyclopedia of the Social & Behavioral Sciences*, pages 621–626.

Chris Collins. 2005. A smuggling approach to the passive in english. *Syntax*, 8(2):81–120.

Chris Collins and Edward Stabler. 2016. A formalization of minimalist syntax. *Syntax*, 19(1):43–78.

Nelson Cowan. 2005. *Working memory capacity.* Psychology Press.

Elizabeth A Cowper. 1976. *Constraints on sentence complexity: a model for syntactic processing.* Ph.D. thesis, Brown University Providence, RI.

Matthew W Crocker and Frank Keller. 2005. Probabilistic grammars as models of gradience in language processing. In *Gradience in grammar: Generative perspectives*.

Aniello De Santo. 2019. Testing a Minimalist grammar parser on Italian relative clause asymmetries. In *Proceedings of the ACL Workshop on Cognitive Modeling and Computational Linguistics (CMCL) 2019*, June 6 2019, Minneapolis, Minnesota.

Aniello De Santo. 2020. Mg parsing as a model of gradient acceptability in syntactic islands. *Proceedings of the Society for Computation in Linguistics*, 3(1):53–63.

Aniello De Santo and Nazila Shafiei. 2019. On the structure of relative clauses in Persian: Evidence from computational modeling and processing effects. In *Talk at the 2nd North American Conference in Iranian Linguistics (NACIL2)*, April 19-21 2019, University of Arizona.

Marica De Vincenzi. 1991. *Syntactic parsing strategies in Italian: The minimal chain principle*, volume 12. Springer Science & Business Media.

Paul Deane. 1991. Limits to attention: A cognitive theory of island phenomena. *Cognitive Linguistics (includes Cognitive Linguistic Bibliography)*, 2(1):1–64.

Vera Demberg and Frank Keller. 2008. Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition*, 109(2):193 – 210.

Vera Demberg and Frank Keller. 2009. A computational model of prediction in human parsing: Unifying locality and surprisal effects. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 31.

Vera Demberg, Frank Keller, and Alexander Koller. 2013. Incremental, predictive parsing with psycholinguistically motivated tree-adjoining grammar. *Computational Linguistics*, 39(4):1025–1066.

Marcel den Dikken. 2000. The syntax of features. *Journal of psycholinguistic research*, 29(1):5–23.

Monica L Do and Elsi Kaiser. 2017. The relationship between syntactic satiation and syntactic priming: A first look. *Frontiers in psychology*, 8:1851.

Shimon Edelman and Morten H Christiansen. 2003. How seriously should we take minimalist syntax? *Trends in Cognitive Sciences*, 7(2):60–61.

Nick C Ellis. 2002. Frequency effects in language processing: A review with implications for theories of implicit and explicit language acquisition. *Studies in second language acquisition*, 24(2):143–188.

Claudia Felser, Colin Phillips, and Matthew Wagers. 2017. Encoding and navigating linguistic representations in memory. *Frontiers in psychology*, 8:164.

Jerry A. Fodor, Bever Thomas G., and Garrett Merrill. 1974. *The psychology of language*. McGraw Hill, New York.

Jerry A. Fodor and Merrill Garrett. 1967. Some syntactic determinants of sentential complexity. *Perception and Psychophysics*, 2:289–296.

Ulrich Hans Frauenfelder, Juan Segui, and Jacques Mehler. 1980. Monitoring around the relative clause. *Journal of Verbal Learning and Verbal Behavior*, 19(3):328–337.

Lyn Frazier. 1978. On comprehending sentences: Syntactic parsing strategies. *Doctoral dissertation, University of Connecticut*.

Lyn Frazier. 1987. Syntactic processing: evidence from Dutch. *Natural Language & Linguistic Theory*, 5(4):519–559.

Naama Friedmann, Adriana Belletti, and Luigi Rizzi. 2009. Relativized relatives: Types of intervention in the acquisition of a-bar dependencies. *Lingua*, 119(1):67–88.

Alan Garnham. 1983. Why psycholinguists don't care about DTC: A reply to Berwick and Weinberg. *Cognition*, 15:263–269.

Carol Georgopoulos. 1985. Variables in palauan syntax. *Natural Language & Linguistic Theory*, 3(1):59–94.

Sabrina Gerth. 2015. *Memory Limitations in Sentence Comprehension: A Structural-based Complexity Metric of Processing Difficulty*, volume 6. Universitätsverlag Potsdam.

Sabrina Gerth and Peter Beim Graben. 2009. Unifying syntactic theory and sentence processing difficulty through a connectionist minimalist parser. *Cognitive neurodynamics*, 3(4):297–316.

Edward Gibson. 1998. Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.

Edward Gibson. 2000. The dependency locality theory: a distance-based theory of linguistic complexity. In *2000, Image, Language, Brain: Papers from the First Mind Articulation Project Symposium*, pages 95–126. MIT press.

Edward Gibson and Evelina Fedorenko. 2010. Weak quantitative standards in linguistics research. *Trends in Cognitive Sciences*, 14(6):233–234.

Edward Gibson, Steven T. Piantadosi, and Evelina Fedorenko. 2013. Quantitative methods in syntax/semantics research: A response to sprouse and almeida (2013). *Language and Cognitive Processes*, 28(3):229–240.

Edward Gibson and James Thomas. 1999. Memory limitations and structural forgetting: The perception of complex ungrammatical sentences as grammatical. *Language and Cognitive Processes*, 14(3):225–248.

Edward Gibson and H-H Iris Wu. 2013. Processing chinese relative clauses in context. *Language and Cognitive Processes*, 28(1-2):125–155.

Helen Goodluck and Susan Tavakolian. 1982. Competence and processing in children's grammar of relative clauses. *Cognition*, 11(1):1–27.

Peter C Gordon, Randall Hendrick, and Marcus Johnson. 2001. Memory interference during language processing. *Journal of experimental psychology: learning, memory, and cognition*, 27(6):1411.

Philip B Gough. 1966. The verification of sentences: The effects of delay of evidence and sentence length. *Journal of Memory and Language*, 5(5):492.

Thomas Graf. 2012. Movement-generalized minimalist grammars. In *International Conference on Logical Aspects of Computational Linguistics*, pages 58–73. Springer.

Thomas Graf. 2013. *Local and Transderivational Constraints in Syntax and Semantics*. Ph.D. thesis, UNIVERSITY OF CALIFORNIA Los Angeles.

Thomas Graf. 2014. Late merge as lowering movement in minimalist grammars. In *International Conference on Logical Aspects of Computational Linguistics*, pages 107–121. Springer.

Thomas Graf, Alëna Aksënova, and Aniello De Santo. 2015a. A single movement normal form for minimalist grammars. In *Formal Grammar*, pages 200–215. Springer.

Thomas Graf and Aniello De Santo. 2019. Sensing tree automata as a model of syntactic dependencies. In *Proceedings of the 16th Meeting on the Mathematics of Language*, pages 12–26.

Thomas Graf, Brigitta Fodor, James Monette, Gianpaul Rachiele, Aunika Warren, and Chong Zhang. 2015b. A refined notion of memory usage for minimalist parsing. In *Proceedings of the 14th Meeting on the Mathematics of Language (MoL 2015)*, pages 1–14, Chicago, USA. Association for Computational Linguistics.

Thomas Graf and Bradley Marcinek. 2014. Evaluating evaluation metrics for minimalist parsing. In *Proceedings of the 2014 ACL Workshop on Cognitive Modeling and Computational Linguistics*, pages 28–36.

Thomas Graf, James Monette, and Chong Zhang. 2017. Relative clauses as a benchmark for Minimalist parsing. *Journal of Language Modelling*, 5:57–106.

Olivia Guest and Andrea E Martin. 2020. How computational modeling can force theory building in psychological science.

Gregory R Guy. 1997. Violable is variable: Optimality theory and linguistic variation. *Language Variation and Change*, 9(3):333–347.

John Hale. 2001. A probabilistic earley parser as a psycholinguistic model. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics*.

John Hale. 2006. Uncertainty about the rest of the sentence. *Cognitive Science*, 30(4):643–672.

John Hale. 2011. What a rational parser would do. *Cognitive Science*, 35(3):399–443.

John Hale. 2016. Information-theoretical complexity metrics. *Language and Linguistics Compass*, 10(9):397–412.

John Hale and Paul Smolensky. 2001. A parser for harmonic context-free grammars. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 23.

Morris Ed Halle, Joan Ed Bresnan, and George A Miller. 1978. *Linguistic theory and psychological reality.* Massachusetts Inst of Technology Pr.

Henk Harkema. 2001. A characterization of minimalist languages. In *International Conference on Logical Aspects of Computational Linguistics*, pages 193–211. Springer.

Philip Hofmeister, Laura Staum Casasanto, and Ivan A Sag. 2012a. How do individual cognitive differences relate to acceptability judgments? A reply to Sprouse, Wagers, and Phillips. *Language*, pages 390–400.

Philip Hofmeister, Laura Staum Casasanto, and Ivan A Sag. 2012b. Misapplying working-memory tests: A reductio ad absurdum. *Language*, 88(2):408–409.

Philip Hofmeister and Ivan A Sag. 2010. Cognitive constraints and island effects. *Language*, 86(2):366–415.

Tim Hunter. 2011. Insertion minimalist grammars: Eliminating redundancies between merge and move. In *Conference on Mathematics of Language*, pages 90–107. Springer.

Tim Hunter. 2018a. Formal methods in experimental syntax. *The Oxford Handbook of Experimental Syntax*.

Tim Hunter. 2018b. Left-corner parsing of Minimalist grammars. In *Minimalist Parsing*. Oxford University Press.

Tim Hunter. 2019. What sort of cognitive hypothesis is a derivational theory of grammar? *Catalan Journal of Linguistics*, 0(0):89–138.

Tim Hunter and Chris Dyer. 2013. Distributions on minimalist grammar derivations. In *Proceedings of the 13th Meeting on the Mathematics of Language (MoL 13)*, pages 1–11.

Tim Hunter and Robert Frank. 2014. Eliminating rightward movement: Extraposition as flexible linearization of adjuncts. *Linguistic Inquiry*, 45(2):227–267.

Tim Hunter, Miloš Stanojević, and Edward Stabler. 2019. The active-filler strategy in a move-eager left-corner minimalist grammar parser. In *Proceedings of the Workshop on Cognitive Modeling and Computational Linguistics*, pages 1–10.

James Hutton and Evan Kidd. 2011. *Structural priming in comprehension of relative clauses: In search of a frequency by regularity interaction.*, pages 227 – 242.

T Florian Jaeger and Neal E Snider. 2013. Alignment as a consequence of expectation adaptation: Syntactic priming is affected by the prime's prediction error given both prior and recent experience. *Cognition*, 127(1):57–83.

Lena A Jäger, Felix Engelmann, and Shravan Vasishth. 2015. Retrieval interference in reflexive processing: experimental evidence from mandarin, and computational modeling. *Frontiers in psychology*, 6:617.

Mark Johnson. 1996. Left corner transforms and finite state approximations. *Tech report MLTT-026, Rank Xerox Research Centre, Grenoble*.

Aravind K Joshi. 1990. Processing crossed and nested dependencies: An automation perspective on the psycholinguistic results. *Language and cognitive processes*, 5(1):1–27.

Aravind K Joshi and Yves Schabes. 1997. Tree-adjoining grammars. In *Handbook of formal languages*, pages 69–123. Springer.

Marcel Adam Just, Patricia A. Carpenter, and Akira Miyake. 2003. Neuroindices of cognitive workload: Neuroimaging, pupillometric and event-related potential studies of brain work. In *Theoretical Issues in Ergonomics Science*, pages 56–88.

Ronald M Kaplan. 1995. Three seductions of computational psycholinguistics. *Formal Issues in Lexical-Functional Grammar*, 47.

Richard S Kayne. 1994. *The antisymmetry of syntax*. 25. MIT Press.

Frank Keller. 2000. *Gradience in grammar: Experimental and computational aspects of degrees of grammaticality*. Ph.D. thesis, The University of Edinburgh.

Gerrit Kentner. 2019. Rhythmic parsing. *The Linguistic Review*, 34(1):123–155.

Jonathan W King and Marta Kutas. 1995. Who did what and when? using word-and clause-level erps to monitor working memory usage in reading. *Journal of cognitive neuroscience*, 7(3):376–395.

Robert Kluender. 1992. Deriving island constraints from principles of predication. In *Island constraints: Theory, acquisition and processing*, pages 223–258. Springer.

Robert Kluender. 1998. On the distinction between strong and weak islands: A processing perspective. In *The limits of syntax*, pages 241–279. Brill.

Robert Kluender and Marta Kutas. 1993. Subjacency as a processing phenomenon. *Language and cognitive processes*, 8(4):573–633.

Robert E Kluender. 1993. *Cognitive constraints on variables in syntax*. Ph.D. thesis, University of California at San Diego.

Gregory M Kobele. 2008. Across-the-board extraction in minimalist grammars. In *Proceedings of the Ninth International Workshop on Tree Adjoining Grammar and Related Frameworks (TAG+ 9)*, pages 113–120.

Gregory M Kobele. 2009. Without remnant movement, mgs are context-free. In *The mathematics of language*, pages 160–173. Springer.

Gregory M Kobele, Sabrina Gerth, and John Hale. 2013. Memory resource allocation in top-down minimalist parsing. In *Formal Grammar*, pages 32–51. Springer.

Gregory M Kobele and Jens Michaelis. 2011. Disentangling notions of specifier impenetrability: Late adjunction, islands, and expressive power. In *Conference on Mathematics of Language*, pages 126–142. Springer.

Gregory M. Kobele, Christian Retoré, and Sylvain Salvati. 2007. An automata-theoretic approach to minimalism. In *Model Theoretic Syntax at 10*, pages 73–82. J. Rogers and S. Kepser.

Gregory Ml Kobele. 2006. *Generating Copies: An investigation into structural identity in language and grammar*. Ph.D. thesis, University of California, Los Angeles.

Annika Kohrt, Trey Sorensen, and Dustin A Chacón. 2018. The real-time status of semantic exceptions to the adjunct island constraint. In *Proceedings of WECOL 2018: Western Conference on Linguistics*.

Dave Kush and Brian Dillon. To appear. Sentence processing and syntactic theory. In *The Blackwell Companion to Chomsky*.

Dave Kush, Terje Lohndal, and Jon Sprouse. 2018. Investigating variation in island effects. *Natural language & linguistic theory*, 36(3):743–779.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2014. Measuring gradience in speakers' grammaticality judgements. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 36.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2015. Unsupervised prediction of acceptability judgements. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, volume 1, pages 1618–1628.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2017. Grammaticality, acceptability, and probability: A probabilistic view of linguistic knowledge. *Cognitive Science*, 41(5):1202–1241.

So Young Lee. 2018. A minimalist parsing account of attachment ambiguity in English and Korean. *Journal of Cognitive Science*, 19(3):291–329.

Geraldine Legendre, Yoshiro Miyata, and Paul Smolensky. 1990. Harmonic grammar–a formal multi-level connectionist theory of linguistic well-formedness: Theoretical foundations. *Proceedings of the Twelfth Annual Conference of the Cognitive Science Society*, page 884?891.

Willem JM Levelt and Andrew Barnas. 1974. *Formal grammars in linguistics and psycholinguistics*, volume 3. Mouton The Hague.

Roger Levy. 2013. Memory and surprisal in human sentence comprehension. In *Sentence processing*, pages 90–126. Psychology Press.

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

Richard L Lewis, Shravan Vasishth, and Julie A Van Dyke. 2006. Computational principles of working memory in sentence comprehension. *Trends in cognitive sciences*, 10(10):447–454.

Shevaun Lewis and Colin Phillips. 2015. Aligning grammatical theories and language processing models. *Journal of Psycholinguistic Research*, 44(1):27–46.

Tal Linzen and Yohei Oseki. 2018. The reliability of acceptability judgments across languages. *Glossa: a journal of general linguistics*, 3(1):100.

Lei Liu. 2018. Minimalist Parsing of Heavy NP Shift. In *Proceedings of PACLIC 32 The 32nd Pacific Asia Conference on Language, Information and Computation*, The Hong Kong Polytechnic University, Hong Kong SAR.

Barbara J Luka and Lawrence W Barsalou. 2005. Structural facilitation: Mere exposure effects for grammatical acceptability as evidence for syntactic priming in comprehension. *Journal of Memory and Language*, 52(3):436–459.

Maryellen C MacDonald, Neal J Pearlmutter, and Mark S Seidenberg. 1994. The lexical nature of syntactic ambiguity resolution. *Psychological review*, 101(4):676.

Kyle Mahowald, Ariel James, Richard Futrell, and Edward Gibson. 2016. A meta-analysis of syntactic priming in language production. *Journal of Memory and Language*, 91:5–27.

David Marr, Tomaso Poggio, Ellen C Hildreth, and W Eric L Grimson. 1991. A computational theory of human stereo vision. In *From the Retina to the Neocortex*, pages 263–295. Springer.

William Marslen-Wilson. 1973. Linguistic structure and speech shadowing at very short latencies. *Nature*, 244(5417):522–523.

William Marslen-Wilson and Lorraine Komisarjevsky Tyler. 1980. The temporal structure of spoken language understanding. *Cognition*, 8(1):1–71.

Paul Marty, Emmanuel Chemla, and Jon Sprouse. 2019. The effect of three basic task features on the sensitivity of acceptability judgment tasks. *Manuscript. https://ling. auf. net/lingbuzz/004588*.

Brian McElree. 2006. Accessing recent events. *Psychology of learning and motivation*, 46:155–200.

Brian McElree, Stephani Foraker, and Lisbeth Dyer. 2003. Memory structures that subserve sentence comprehension. *Journal of memory and language*, 48(1):67–91.

Jens Michaelis. 1998. Derivational minimalism is mildly context–sensitive. In *International Conference on Logical Aspects of Computational Linguistics*, pages 179–198. Springer.

George A. Miller and Noam Chomsky. 1963. Finitary models of language users. In R. Luce, R. Bush, and E. Galanter, editors, *Handbook of Mathematical Psychology*, volume 2. John Wiley, New York.

George A. Miller and Kathryn Ojemann McKean. 1964. A chronometric study of some relations between sentences. *Quarterly Journal of Experimental Psychology*, 16:297–308.

Bruno Nicenboim and Shravan Vasishth. 2018. Models of retrieval in sentence comprehension: A computational evaluation using bayesian hierarchical modeling. *Journal of Memory and Language*, 99:1 – 34.

Bruno Nicenboim, Shravan Vasishth, Carolina Gattei, Mariano Sigman, and Reinhold Kliegl. 2015. Working memory differences in long-distance dependency resolution. *Frontiers in Psychology*, 6:312.

William O'Grady. 2011. Relative clauses. processing and acquisition. In Evan Kidd, editor, *Processing, Typology, and Function*, pages 13–38. John Benjamins, Amsterdam.

Isabel Oltra-Massuet, Victoria Sharpe, Kyriaki Neophytou, and Alec Marantz. 2017. Syntactic priming as a test of argument structure: A self-paced reading experiment. *Frontiers in psychology*, 8:1311.

Francisco Ordóñez. 1998. Post-verbal asymmetries in Spanish. *Natural Language & Linguistic Theory*, 16(2):313–345.

Dan Parker, Michael Shvartsman, and Julie A Van Dyke. 2017. The cue-based retrieval theory of sentence comprehension: New findings and new challenges. *Language processing and disorders*, pages 121–144.

Robert Pasternak and Thomas Graf. 2020. Cyclic scope and processing difficulty in a minimalist parser. *Manuscript. https://ling.auf.net/lingbuzz/005009*.

Joe Pater. 2008. Gradual learning and convergence. *Linguistic Inquiry*, 39(2):334–345.

Joe Pater. 2009. Weighted constraints in generative linguistics. *Cognitive science*, 33(6):999–1035.

Joe Pater. 2016. Universal grammar with weighted constraints. *Harmonic grammar and harmonic serialism (2016): 1-46.*, pages 1–46.

Colin Phillips. 1996. *Order and Structure*. Ph.D. thesis, MIT.

Colin Phillips. 2003. Parsing: Psycholinguistic aspects. In *International Encyclopedia of Linguistics*, 2 edition. Oxford University Press.

Colin Phillips. 2009. Should we impeach armchair linguists? *Japanese/Korean Linguistics*, 17:49–64.

Colin Phillips. 2013a. On the nature of island constraints i: language processing and reductionist accounts. *Experimental syntax and island effects*, pages 64–108.

Colin Phillips. 2013b. On the nature of island constraints ii: Language learning and innateness. *Experimental syntax and island effects*, pages 132–157.

Colin Phillips and Howard Lasnik. 2003. Linguistics and empirical evidence. reply to edelman and christiansen. *Trends in cognitive sciences*, 7(2):61–62.

Martin J Pickering and Holly P Branigan. 1998. The representation of verbs: Evidence from syntactic priming in language production. *Journal of Memory and language*, 39(4):633–651.

Martin J Pickering and Victor S Ferreira. 2008. Structural priming: A critical review. *Psychological bulletin*, 134(3):427.

Till Poppels and Andrew Kehler. 2019. Reconsidering asymmetries in voice-mismatched vp-ellipsis. *Glossa: a journal of general linguistics*, 4(1).

Mary C Potter and Linda Lombardi. 1998. Syntactic priming in immediate recall of sentences. *Journal of Memory and Language*, 38(3):265–282.

Christopher Potts, Joe Pater, Karen Jesney, Rajesh Bhatt, and Michael Becker. 2010. Harmonic grammar with linear programming: from linear systems to linguistic typology. *Phonology*, 27(1):77–117.

Alan Prince and Paul Smolensky. 2008. *Optimality Theory: Constraint interaction in generative grammar*. John Wiley & Sons.

Owen Rambow and Aravind K Joshi. 1994. A processing model for free word order languages. *Perspectives on Sentence Processing*.

Florencia Reali and Morten H Christiansen. 2007. Word chunk frequencies affect the processing of pronominal object-relative clauses. *The Quarterly Journal of Experimental Psychology*, 60(2):161–170.

David Reitter, Frank Keller, and Johanna D Moore. 2011. A computational cognitive model of syntactic priming. *Cognitive science*, 35(4):587–637.

Philip Resnik. 1992. Left-corner parsing and psychological plausibility. In *Proceedings of the 14th conference on Computational linguistics-Volume 1*, pages 191–197. Association for Computational Linguistics.

Luigi Rizzi. 1980. Violations of the wh island constraint in italian and the subjacency condition. *Journal of Italian Linguistics Amsterdam*, 5(1):157–191.

Luigi Rizzi. 1990. *Relativized minimality.* The MIT Press.

Corianne Rogalsky and Gregory Hickok. 2008. Selective attention to semantic and syntactic features modulates sentence processing networks in anterior temporal cortex. *Cerebral Cortex*, 19(4):786–796.

Iris van Rooij and Giosuè Baggio. 2020. Theory before the test: How to build high-verisimilitude explanatory theories in psychological science.

Daniel J Rosenkrantz and Philip M Lewis. 1970. Deterministic left corner parsing. In *11th Annual Symposium on Switching and Automata Theory (swat 1970)*, pages 139–152. IEEE.

J. R. Ross. 1968. Constraints on variables in syntax. *Ph.D. dissertation, MIT*.

Kofi K Saah and Helen Goodluck. 1995. Island effects in parsing and grammar: Evidence from akan. *The Linguistic Review*, 12(4):381–410.

Joachim Sabel. 2002. A minimalist analysis of syntactic islands. *Linguistic Review*, 19(3):271–315.

Sylvain Salvati. 2011. Minimalist grammars in the light of logic. In *Logic and grammar*, pages 81–117. Springer.

Harris B Savin and Ellen Perchonock. 1965. Grammatical structure and the immediate recall of english sentences. *Journal of Memory and Language*, 4(5):348.

Herbert Schriefers, Angela D Friederici, and Katja Kuhn. 1995. The processing of locally ambiguous relative clauses in german. *Journal of Memory and Language*, 34(4):499.

Carson T Schütze. 2011. Linguistic evidence and grammatical theory. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(2):206–221.

Carson T Schütze. 2016. *The empirical base of linguistics: Grammaticality judgments and linguistic methodology*.

Nazila Shafiei and Thomas Graf. 2020. The subregular complexity of syntactic islands. *Proceedings of the Society for Computation in Linguistics*, 3(1):272–281.

Klaas Sikkel. 2012. *Parsing Schemata: A Framework for Specification and Analysis of Parsing Algorithms*. Springer Science & Business Media.

Dan Slobin. 1966. Grammatical transformations and sentence comprehension in childhood and adulthood. *Journal of Verbal Learning and Verbal Behavior*, 5:219–227.

Antonella Sorace and Frank Keller. 2005. Gradience in linguistic data. *Lingua*, 115(11):1497–1524.

Jon Sprouse. 2007. Continuous acceptability, categorical grammaticality, and experimental syntax. *Biolinguistics*, 1:123–134.

Jon Sprouse and Diogo Almeida. 2012. Assessing the reliability of textbook data in syntax: Adger's core syntax. *Journal of Linguistics*, 48(3):609–652.

Jon Sprouse and Diogo Almeida. 2017. Design sensitivity and statistical power in acceptability judgment experiments. *Glossa*, 2(1):1.

Jon Sprouse, Ivano Caponigro, Ciro Greco, and Carlo Cecchetto. 2016. Experimental syntax and the variation of island effects in english and italian. *Natural Language & Linguistic Theory*, 34(1):307–344.

Jon Sprouse, Carson T Schütze, and Diogo Almeida. 2013. A comparison of informal and formal acceptability judgments using a random sample from linguistic inquiry 2001–2010. *Lingua*, 134:219–248.

Jon Sprouse, Matt Wagers, and Colin Phillips. 2012a. A test of the relation between working-memory capacity and syntactic island effects. *Language*, 88(1):82–123.

Jon Sprouse, Matt Wagers, and Colin Phillips. 2012b. Working-memory capacity and island effects: A reminder of the issues and the facts. *Language*, 88(2):401–407.

Jon Sprouse, Beracah Yankama, Sagar Indurkhya, Sandiway Fong, and Robert C Berwick. 2018. Colorless green ideas do sleep furiously: Gradient acceptability and the nature of the grammar. *The Linguistic Review*, 35(3):575–599.

Edward P. Stabler. 1984. Berwick and weinberg on linguistics and computational psychology. *Cognition*, 17(2):155–179.

Edward P Stabler. 1996. Derivational minimalism. In *International Conference on Logical Aspects of Computational Linguistics*, pages 68–95. Springer.

Edward P Stabler. 2011. Computational perspectives on minimalism. In *The Oxford Handbook of Linguistic Minimalism*.

Edward P Stabler. 2013. Two models of minimalist, incremental syntactic analysis. *Topics in cognitive science*, 5(3):611–633.

Miloš Stanojević and Edward Stabler. 2018. A sound and complete left-corner parsing for minimalist grammars. In *Proceedings of the Eight Workshop on Cognitive Aspects of Computational Language Learning and Processing*, pages 65–74.

Mark Steedman. 2001. *The Syntactic Process*. MIT Press, Cambridge, MA.

Patrick Sturt, Frank Keller, and Amit Dubey. 2010. Syntactic priming in comprehension: Parallelism effects with and without coordination. *Journal of Memory and Language*, 62(4):333–351.

Anna Szabolcsi, Martin Everaert, and Henk van Riemsdijk. 2006. The blackwell companion to syntax. *by Henk van Riemsdijk Martin Everaert*, 1:479–531.

Anna Szabolcsi and Terje Lohndal. 2017. Strong vs. weak islands. *The Wiley Blackwell Companion to Syntax, Second Edition*, pages 1–51.

Whitney Tabor. 2009. A dynamical systems perspective on the relationship between symbolic and non-symbolic computation. *Cognitive neurodynamics*, 3(4):415–427.

Michael K Tanenhaus, Michael J Spivey-Knowlton, Kathleen M Eberhard, and Julie C Sedivy. 1995. Integration of visual and linguistic information in spoken language comprehension. *Science*, 268(5217):1632–1634.

James W. Thatcher. 1967. Characterizing derivation trees of context-free grammars through a generalization of finite automata theory. *Journal of Computer and System Sciences*, 1(4):317–322.

Malathi Thothathiri and Jesse Snedeker. 2008. Give and take: Syntactic priming during spoken language comprehension. *Cognition*, 108(1):51–68.

Kristen M Tooley and Matthew J Traxler. 2010. Syntactic priming effects in comprehension: A critical review. *Language and Linguistics Compass*, 4(10):925–937.

John Torr. 2017. Autobank: a semi-automatic annotation tool for developing deep minimalist grammar treebanks. In *Proceedings of the Software Demonstrations of the 15th Conference of the European Chapter of the Association for Computational Linguistics*, pages 81–86, Valencia, Spain. Association for Computational Linguistics.

John Torr. 2018. Constraining mgbank: Agreement, l-selection and supertagging in minimalist grammars. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 590–600.

John Torr, Milos Stanojevic, Mark Steedman, and Shay B Cohen. 2019. Wide-coverage neural a* parsing for minimalist grammars. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 2486–2505.

David J. Townsend and Thomas G. Bever. 2001. *Sentence comprehension: The integration of habits and rules*. MIT Press, Cambridge, MA.

Matthew J Traxler and Martin J Pickering. 1996. Plausibility and the processing of unbounded dependencies: An eye-tracking study. *Journal of Memory and Language*, 35(3):454–475.

Matthew J Traxler, Kristen M Tooley, and Martin J Pickering. 2014. Syntactic priming during sentence comprehension: Evidence for the lexical boost. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 40(4):905.

Melissa Troyer, Timothy J O'Donnell, Evelina Fedorenko, and Edward Gibson. 2011. Storage and computation in syntax: Evidence from relative clause priming. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 33.

Robert Truswell. 2007. Extraction from adjuncts and the structure of events. *Lingua*, 117(8):1355–1377.

Robert Truswell. 2011. *Events, phrases, and questions*. 33. Oxford University Press.

Irene Utzeri. 2007. The production and acquisition of subject and object relative clauses in Italian: a comparative experimental study. *Nanzan Linguistics*, 2.

Julie A Van Dyke and Richard L Lewis. 2003. Distinguishing effects of structure and decay on attachment and repair: A cue-based parsing account of recovery from misanalyzed ambiguities. *Journal of Memory and Language*, 49(3):285–316.

Julie A Van Dyke and Brian McElree. 2006. Retrieval interference in sentence comprehension. *Journal of Memory and Language*, 55(2):157–166.

Sandra Villata, Whitney Tabor, and Julie Franck. 2018. Encoding and retrieval interference in sentence comprehension: Evidence from agreement. *Frontiers in psychology*, 9:2.

Francesca Volpato. 2010. *The acquisition of relative clauses and phi-features: evidence from hearing and hearing-impaired populations*. Ph.D. thesis, Università Ca'Foscari di Venezia.

Francesca Volpato and Flavia Adani. 2009. The subject/object relative clause asymmetry in Italian hearing-impaired children: evidence from a comprehension task. *Studies in Linguistics*, 3:269–281.

Matthew W Wagers and Colin Phillips. 2009. Multiple dependencies and the role of the grammar in real-time comprehension. *Journal of Linguistics*, 45(2):395–433.

Heinz Wanner and Michael P. Maratsos. 1978. An ATN approach to comprehension. In *Linguistic theory and psychological reality*. MIT Press.

Victor H Yngve. 1960. A model and an hypothesis for language structure. *Proceedings of the American philosophical society*, 104(5):444–466.

Jiwon Yun, Zhong Chen, Tim Hunter, John Whitman, and John Hale. 2015. Uncertainty in processing relative clauses across East Asian languages. *Journal of East Asian Linguistics*, 24(2):113–148.

Chong Zhang. 2017. *Stacked Relatives: Their Structure, Processing and Computation*. Ph.D. thesis, State University of New York at Stony Brook.

J Ziegler, J Snedeker, and E Wittenberg. 2017. Priming is swell, but it's far from simple. *The Behavioral and brain sciences*, 40:e312–e312.

Jayden Ziegler and Jesse Snedeker. 2018. How broad are thematic roles? evidence from structural priming. *Cognition*, 179:221–240.

Appendices

Appendix A

Metric	*SRC* < *ORC*	*SRC* < *ORCp*	*ORC* < *ORCp*
AvgS	✓	✓	✓
AvgS′	✓	✓	✓
AvgT	✓	✓	✓
AvgT′	✓	✓	✓
BoxT	✓	✓	✓
BoxT′	Tie	✓	✓
MaxS	✓	✓	✓
MaxS′	✓	✓	✓
MaxSR	✓	✓	✓
MaxSR′	✓	✓	✓
MaxT	✓	✓	✓
MaxT′	✓	✓	✓
MaxTR	✓	✓	✓
MaxTR′	✓	✓	✓
Movers	✓	✓	✓
Movers′	Tie	✓	✓
SumS	✓	✓	✓
SumS′	✓	✓	✓
SumT	✓	✓	✓
SumT′	✓	✓	✓

Table A.1: Performance of base metrics for the Italian right-embedding RCs contrasts.

Metric	$SRC < ORC$	$SRC < ORCp$	$ORC < ORCp$
AvgS	No	✓	✓
AvgS′	✓	✓	✓
AvgT	✓	✓	✓
AvgT′	✓	✓	✓
BoxT	✓	✓	✓
BoxT′	Tie	✓	✓
MaxS	Tie	✓	✓
MaxS′	Tie	✓	✓
MaxSR	✓	✓	✓
MaxSR′	✓	✓	✓
MaxT	Tie	✓	✓
MaxT′	Tie	✓	✓
MaxTR	✓	✓	✓
MaxTR′	✓	✓	✓
Movers	✓	✓	✓
Movers′	Tie	✓	✓
SumS	✓	✓	✓
SumS′	✓	✓	✓
SumT	✓	✓	✓
SumT′	✓	✓	✓

Table A.2: Performance of base metrics for the Italian left-embedding RCs contrasts.

Metric	$SVO < VS$	$VS_{\text{unacc}} < VS_{\text{unerg}}$
AvgS	Tie	✓
AvgS′	Tie	✓
AvgT	✓	✓
AvgT′	✓	✓
BoxT	✓	✓
BoxT′	✓	✓
MaxS	✓	✓
MaxS′	✓	✓
MaxSR	✓	✓
MaxSR′	✓	✓
MaxT	✓	✓
MaxT′	✓	✓
MaxTR	✓	✓
MaxTR′	✓	✓
Movers	✓	✓
Movers′	✓	✓
SumS	✓	✓
SumS′	✓	✓
SumT	✓	✓
SumT′	✓	✓

Table A.3: Performance of base metrics for the $SVO < VS$ and Unaccusative $VS <$ Unergative VS contrasts in Italian.

Appendix B

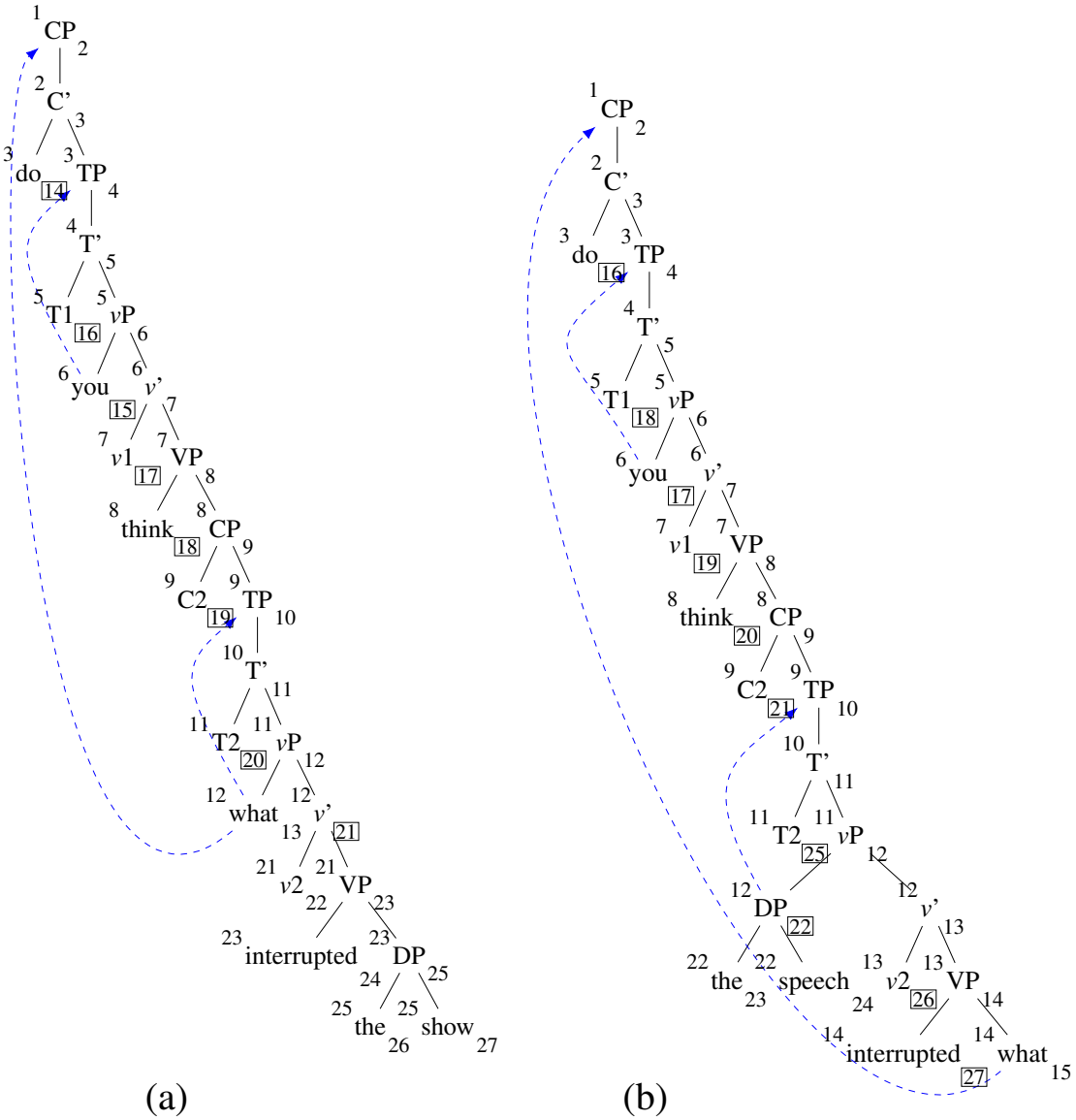


Figure B.1: Annotated derivation trees for the Subject island - case 2 sentences: (a) 24a (Short/Non Island) and (b) 24b (Long/Non Island).

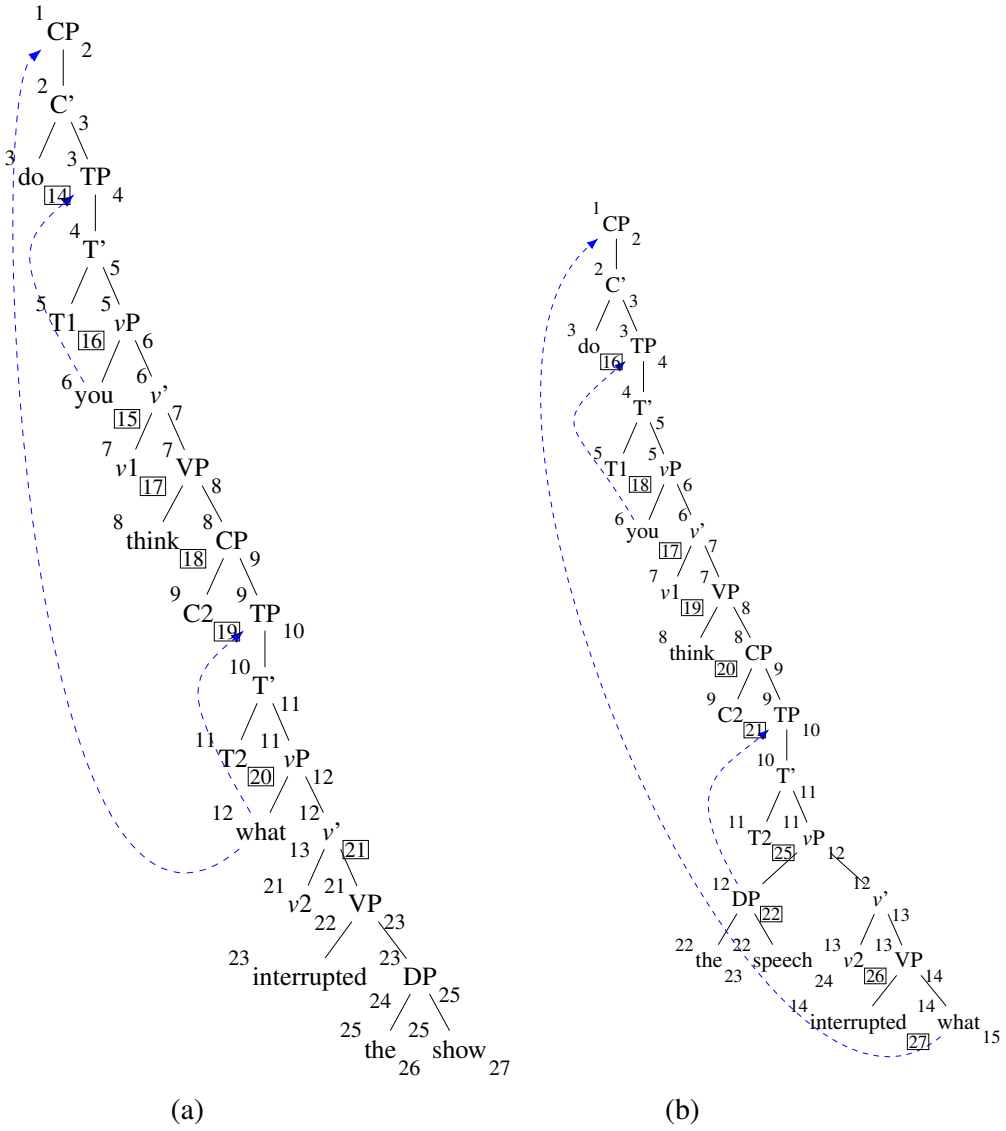


Figure B.2: Annotated derivation trees for the Subject island - case 2 sentences: (a) 24c (Short/ Island) and (b) 24d (Long/ Island).

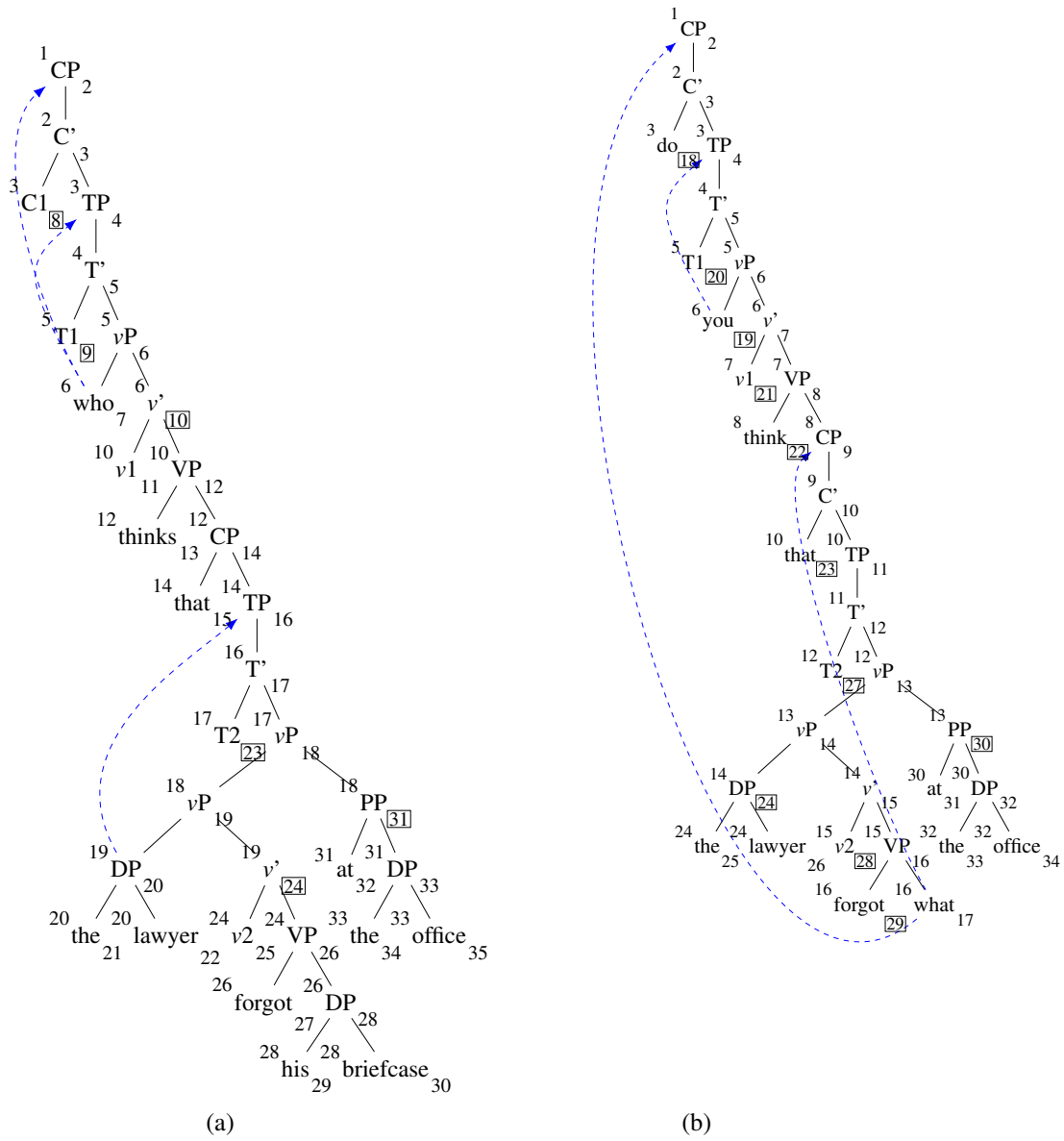


Figure B.3: Annotated derivation trees for the Adjunct island sentences: (a) 25a (Short/ Non Island) and (b) 25b (Long/Non Island).

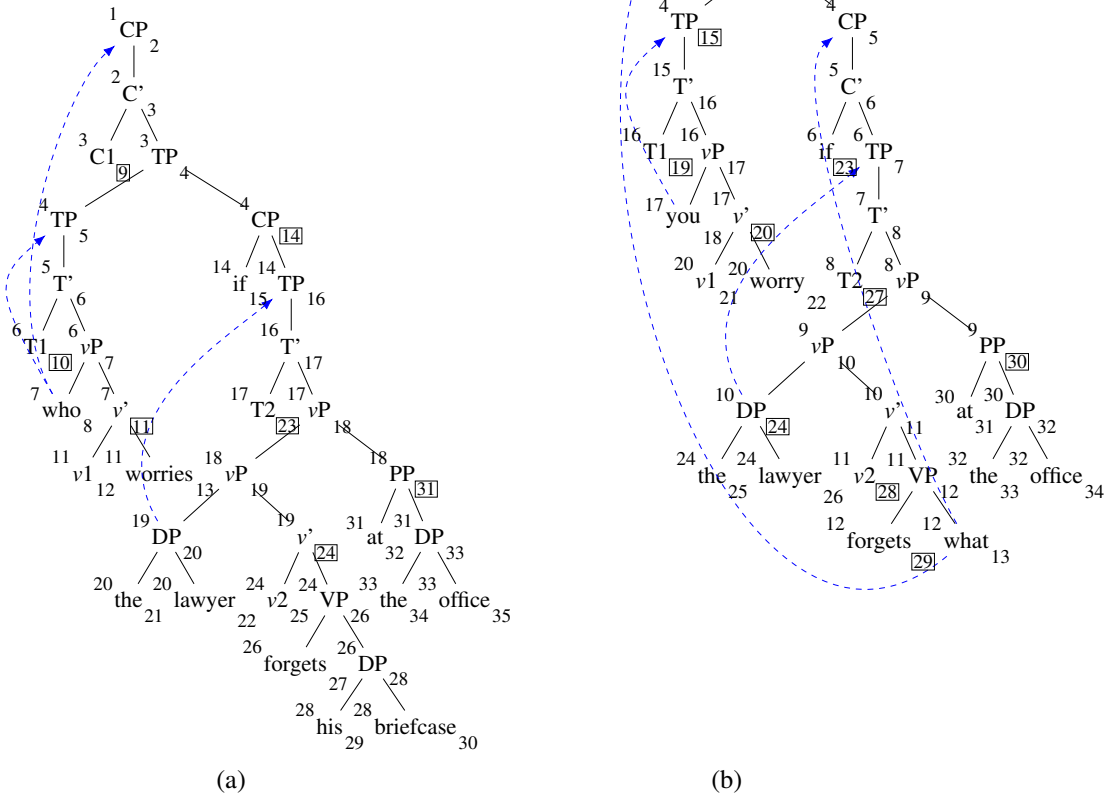


Figure B.4: Annotated derivation trees for the Adjunct island sentences: (a) 25c (Short/ Island) and (b) 25d (Long/ Island).

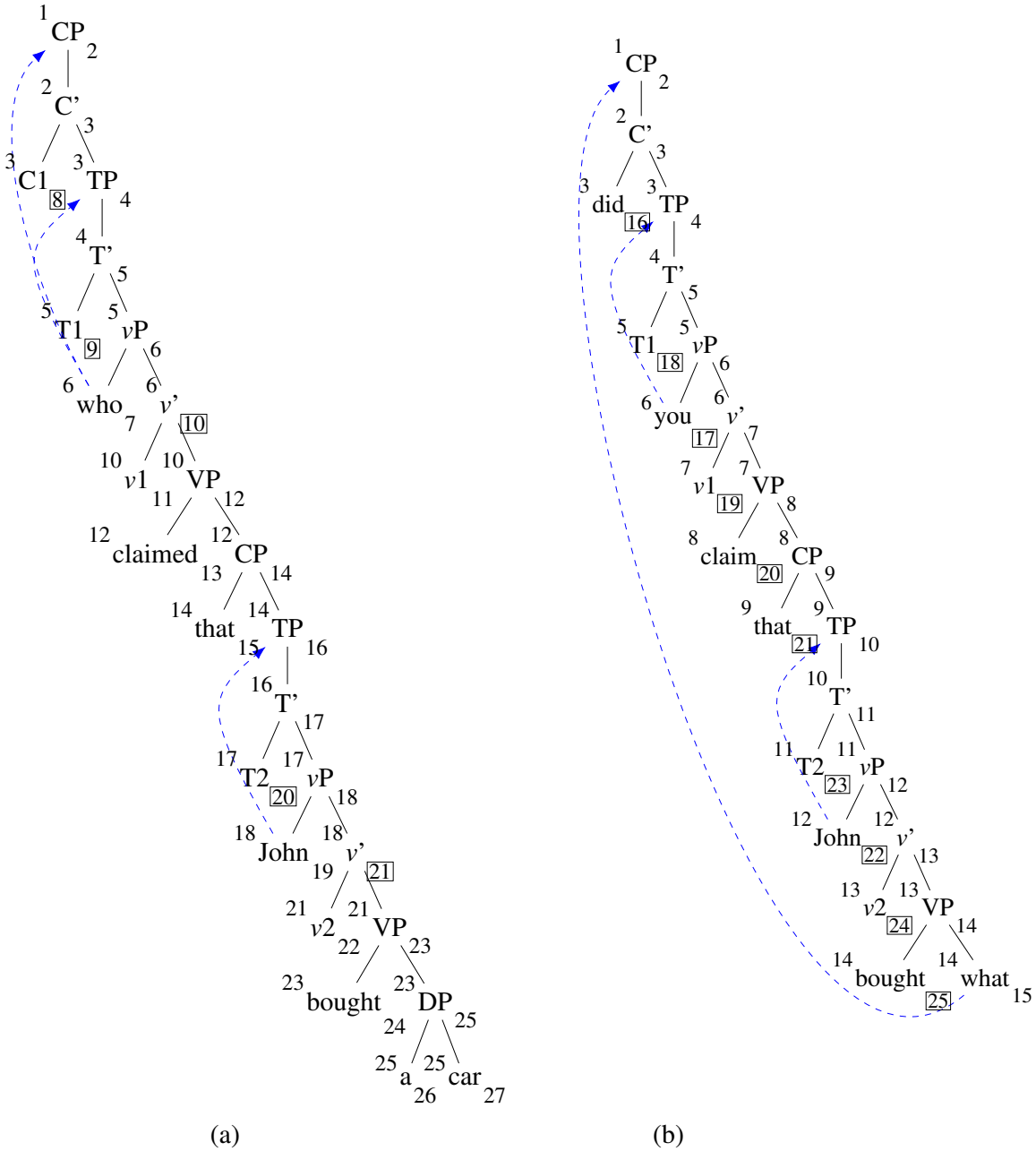


Figure B.5: Annotated derivation trees for the Complex NP island sentences: (a) 26a (Short/ Non Island) and (b) 26b (Long/Non Island).

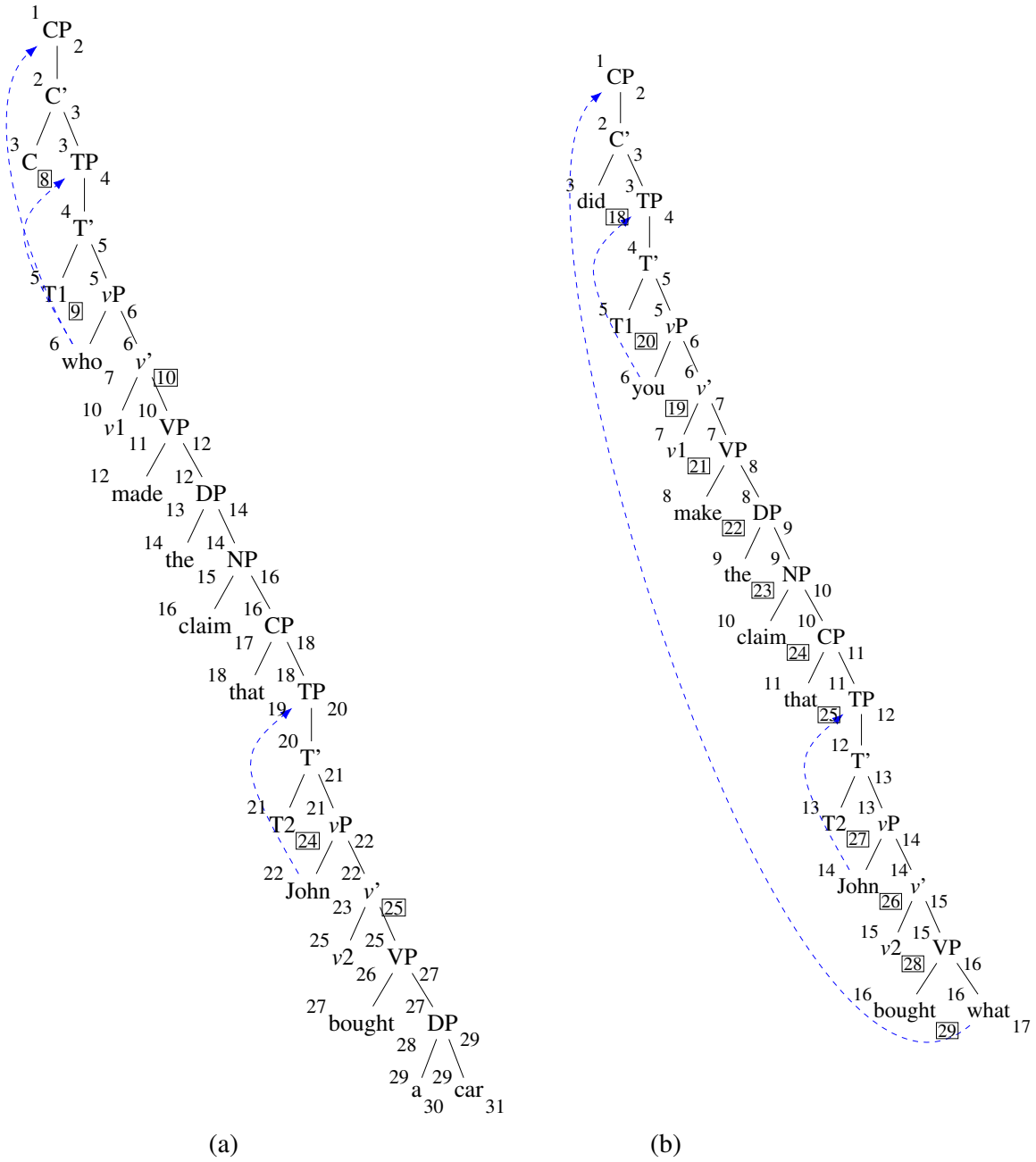


Figure B.6: Annotated derivation trees for the Complex NP island sentences: (a) 26c (Short/ Island) and (b) 26d (Long/ Island).

Metric	Obj. | Non Isl. > Obj. | Isl.	Subj. | Non Isl. > Subj. | Isl.	Subj. | Non Isl. > Obj. | Non Isl.	Obj. | Isl. > Subj. | Isl.	Obj. | Non Isl. > Subj. | Isl.	Subj | Non Isl. > Obj. | Isl.
AvgS	✓	×	✓	×	✓	×
AvgS'	✓	✓	×	×	✓	×
AvgT	✓	✓	×	×	✓	×
AvgT'	✓	✓	×	×	×	×
BoxT	✓	✓	×	×	✓	×
BoxT'	✓	✓	Tie	Tie	✓	×
MaxS	✓	✓	×	×	✓	×
MaxS'	✓	✓	×	×	✓	×
MaxSR	✓	✓	×	×	✓	×
MaxSR'	✓	✓	×	×	✓	×
MaxT	✓	✓	×	×	✓	×
MaxT'	✓	✓	×	×	✓	×
MaxTR	✓	✓	×	×	✓	×
MaxTR'	✓	✓	×	×	✓	×
Movers	Tie	✓	×	Tie	Tie	×
Movers'	Tie	Tie	Tie	Tie	Tie	Tie
SumS	✓	✓	×	×	✓	×
SumS'	✓	✓	×	×	✓	×
SumT	✓	✓	×	×	✓	×
SumT'	✓	✓	×	×	✓	×

Table B.1: Performance of base metrics for each contrast in the Subject Island case 1

Metric	Emb. | Non Isl. > Emb. | Isl.	Matrix | Non Isl. > Matrix | Isl.	Matrix | Non Isl. > Emb. | Non Isl.	Matrix | Non Isl. > Emb. | Isl.	Matrix | Isl. > Emb. | Isl.	Matrix | Isl. > Emb. | Non Isl.
AvgS	×	Tie	✓	✓	✓	✓
AvgS'	✓	Tie	✓	✓	✓	✓
AvgT	✓	✓	✓	✓	✓	✓
AvgT'	✓	✓	✓	✓	✓	✓
BoxT	✓	Tie	✓	✓	✓	✓
BoxT'	✓	✓	×	✓	×	×
MaxS	✓	Tie	✓	✓	✓	✓
MaxS'	✓	Tie	✓	✓	✓	✓
MaxSR	✓	Tie	✓	✓	✓	✓
MaxSR'	✓	Tie	✓	✓	✓	✓
MaxT	✓	✓	✓	✓	✓	Tie
MaxT'	✓	✓	✓	✓	✓	Tie
MaxTR	✓	✓	✓	✓	✓	×
MaxTR'	✓	✓	✓	✓	✓	×
Movers	✓	Tie	Tie	✓	✓	Tie
Movers'	Tie	Tie	Tie	Tie	Tie	Tie
SumS	✓	Tie	✓	✓	✓	✓
SumS'	✓	Tie	✓	✓	✓	✓
SumT	✓	✓	✓	✓	✓	✓
SumT'	✓	✓	✓	✓	✓	✓

Table B.2: Performance of base metrics for each contrast in the Subject Island case 2

Metric	Matrix | Non Isl. > Matrix | Isl.	Matrix | Non Isl. > Emb. | Non Isl.	Matrix | Non Isl. > Emb. | Isl.	Matrix | Isl. > Emb. | Isl.	Matrix | Isl. > Emb. | Non Isl.	Emb. | Non Isl. > Emb. | Isl.
AvgS	✓	✓	✓	✓	✓	✓
AvgS'	✓	✓	✓	✓	✓	✓
AvgT	✓	✓	✓	✓	✓	×
AvgT'	✓	✓	✓	✓	✓	×
BoxT	✓	✓	✓	✓	✓	×
BoxT'	Tie	×	×	×	×	Tie
MaxS	✓	✓	✓	✓	✓	×
MaxS'	✓	✓	✓	✓	✓	×
MaxSR	✓	✓	✓	✓	✓	×
MaxSR'	✓	✓	✓	✓	✓	×
MaxT	Tie	✓	✓	✓	✓	✓
MaxT'	Tie	✓	✓	✓	✓	✓
MaxTR	✓	✓	✓	✓	✓	✓
MaxTR'	✓	✓	✓	✓	✓	✓
Movers	Tie	Tie	✓	✓	Tie	✓
Movers'	Tie	Tie	✓	✓	Tie	✓
SumS	✓	✓	✓	✓	✓	✓
SumS'	✓	✓	✓	✓	✓	✓
SumT	✓	✓	✓	✓	✓	×
SumT'	✓	✓	✓	✓	✓	×

Table B.3: Performance of base metrics for each contrast in the Adjunct Island case

Metric	Matrix | Non Isl. = Matrix | Isl.	Matrix | Non Isl. > Emb. | Non Isl.	Matrix | Non Isl. > Emb. | Isl.	Emb. | Non Isl. > Emb. | Isl.	Matrix | Isl. > Emb. | Non Isl.	Matrix | Isl. > Emb. | Isl.
AvgS	Tie	✓	✓	✓	✓	✓
AvgS'	Tie	✓	✓	✓	✓	✓
AvgT	Tie	✓	✓	✓	✓	✓
AvgT'	×	✓	✓	✓	✓	✓
BoxT	Tie	✓	✓	✓	✓	✓
BoxT'	✓	×	✓	✓	×	×
MaxS	Tie	✓	✓	✓	✓	✓
MaxS'	Tie	✓	✓	✓	✓	✓
MaxSR	Tie	✓	✓	✓	✓	✓
MaxSR'	Tie	✓	✓	✓	✓	✓
MaxT	Tie	✓	✓	✓	✓	✓
MaxT'	Tie	✓	✓	✓	✓	✓
MaxTR	Tie	✓	✓	✓	✓	✓
MaxTR'	✓	✓	✓	✓	✓	✓
Movers	Tie	✓	✓	Tie	✓	✓
Movers'	Tie	Tie	Tie	Tie	Tie	Tie
SumS	Tie	✓	✓	✓	✓	✓
SumS'	Tie	✓	✓	✓	✓	✓
SumT	Tie	✓	✓	✓	✓	✓
SumT'	✓	✓	✓	✓	✓	✓

Table B.4: Performance of base metrics for each contrast in the Complex NP Island case