UNIVERSITAS BERGENSIS

Department of Linguistic, Literary and Aesthetic Studies

DASP350 - Natural Language Processing

Master's Thesis - Fall 2013

A psycholinguistically plausible alternative to the standard pipeline in Natural Language Processing

---

By: Magnus Bakken

Supervisor: Christer Johansson

Co-supervisor: Lars G. Johnsen

Acknowledgements

I would like to thank Uwe Wolter for guiding me in the direction of category theory and coalgebras, for taking time out of his busy schedule for helping me get started with the project and providing feedback along the way.

I am grateful to my supervisors Christer Johansson and Lars G. Johnsen for allowing me to explore this strange combination of topics, and for their valuable feedback on the project and the text.

Thanks to Antal van den Bosch for his courteous and substantial replies to my questions. Thanks to Seong-Eun Cho, Truls Pedersen, Hannah Hansen, Lubos Steskal and Kjetil Golid, who have provided valuable feedback on the project.

Finally, I want to thank my partner Helene for being supportive and patient throughout the process.

Abstract

Research in Natural Language Processing has assumed that high level tasks such as Natural Language Understanding can be solved by special purpose processes operating in a pipeline, called the standard pipeline. Work is then done on building the individual components. However, there appears to be no way of guaranteeing that such components will actually solve high level NLP tasks when combined as suggested by the standard pipeline.

Human beings represent the only working mechanism that can solve high level NLP tasks, but human language comprehension does not correspond to the standard pipeline. I attempt to provide a psycholinguistically motivated alternative to the standard pipeline. That is, an alternative architechture for the process of language understanding. I focus on modeling structural properties of the process rather than implementation details.

Coalgebra is a mathematical formalism that allows structural properties of different types of processes to be described mathematically (Rutten, 2000), and is employed here in order to capture central structural properties of human language comprehension.

In the alternative model syntax- and meaning representations are incrementally constructed on a word-by-word basis. Processing is serial in that the model commits to a unique syntactic and semantic increment for each word. Incrementally constructed structures are stored in states that both result from- and influence processing. Meaning representations do not influence further syntactic increments, although meaning representations can cause the process to throw an error. I propose a reanalysis algorithm by which the process can recover from such errors.

Additionally, the states of the model include representations of topic level context through states of knowledge activation. These are important in deciding how to increment the meaning representation. Finally, context models are important in refining when the model throws a meaning related error. These are states of the model that represent the events, relations and entities described thus far in preceding text. Context models are important in making the model less brittle when processing some types of figurative language.

Sammendrag

Forskning på prossesering av naturlig språk har antatt at høynivå-oppgaver som forståelse av naturlig språk kan løses ved at oppgaver som prosseseres ulike typer språklig informasjon kobles sekvensielt. Denne arkitekturen kalles “the standard pipeline”. Man arbeider så med å lage komponentene som inngår i sekvensen. Det finnes derimot ingen garanti for at komponentene til sammen vil kunne forstå språk når de er koblet i den foreslåtte sekvensen.

Mennesket representerer den eneste kjente mekanismen som klarer å løse slike høynivå-oppgaver, men menneskets språkforståelse følger ikke the standard pipeline. I oppgaven forsøker jeg å lage en modell av arkitekturen på prosessen som mennesker bruker for å forstå språk, motivert av psykolingvistisk forskning. Jeg fokuserer på arkitekturen av prosessen, fremfor implementasjonsdetaljer.

Coalgebra er en formalisme som kan representere ulike strukturelle aspekter ved prosseser matematisk (Rutten, 2000), og er brukt i oppgaven for å representere strukturelle egenskaper ved menneskelig språkforståelse.

I den alternative modellen blir representasjoner av syntaks of betydning inkrementelt konstruert, ord for ord. Prosesseringen er seriell, da den forplikter seg til unike syntaktiske og semantiske inkrement etter hvert ord. Inkrementelt konstruerte strukturer er lagret i tilstander, som både resulterer fra- og påvirker videre prosessering. Betydnings-representasjoner påvirker imidlertid ikke videre syntaktisk prossesering direkte, men kan forårsake at prossesen hever et unntak. I oppgaven foreslår jeg en algoritme for å velge en annen analyse når prosessen hever et unntak.

I tillegg inneholder tilstandene i modellen representasjoner av kontekst på emnenivå gjennom tilstander av aktivert kunnskap. Disse tilstandene er viktige for å bestemme hvordan betydningsrepresentasjonen skal bygges inkrementelt, og kan også påvirke inkrementelle syntaktiske valg. Tilstandene i modellen inneholder også representasjoner av hendelsene, relasjonene og entitetene beskrevet så langt i en tekst i kontekst-modeller. Kontekst-modeller er viktige for å raffinere tilfellene hvor modellen skal heve en betydningsrelatert feil. Kontekst-modeller gjør modellen mer robust når det gjelder å prossesere noen typer figurativt språk.

Contents

Chapter 1

Introduction

The purpose of this chapter is to introduce the background, motivation, goals and methods of the project. The background for the present project is that we do not yet know how to assemble a program that solves natural language understanding robustly (section 1.1). Psycholinguistics provides information about a working design (human beings) that could inspire working programs. The main goal of the project is to come up with a mathematical model of how humans process language anchored in findings from psycholinguistics (section 1.2). This model should serve as a high level architechture for programs that process language. Secondarily, I seek to show how the model can make headway on some of the problems that have faced programs constructed according to the traditional architechture (to be introduced).

1.1 Background

A central goal in natural language processing (NLP) is to engineer systems that understand natural language. This problem is called natural language understanding and involves assigning meaning representations to natural language. The problem of natural language understanding is thought to be among the hardest problems to solve in all of Artificial Intelligence, because it contains many of the problems an intelligent entity should be able to handle (Turing, 1950). The ability to use and represent world knowledge is conjectured to be necessary for natural language understanding.

Data sparseness suggests that natural language understanding must be decomposed into tasks (with lower individual data sparseness) that combine to process individual sentences. That is, there must be some inner architechure of subtasks that are combined in a consistent way in order to accomplish natural language understanding.

A large array of such NLP subtasks exist. Work in the field of NLP is often concentrated around solving these subtasks. Comparatively little work is done on how such computational components ought to combine in large-scale applications. A traditional view of this architecture (to be discussed in chapter 2) is the *the standard pipeline of natural language processing*.

It is in general quite difficult to tell in advance whether or not an architechture of processes will be successful at natural language understanding once the component processes are constructed and assembled in the prescribed way. This problem also affects the standard pipeline. Human beings represent the only known mechanism to accomplish natural language understanding. Humans comprehend language in a flexible way, that can be extended to cover multiple domains. We take context and knowledge into account when comprehending sentences. Figurative language differs from literal language in that its meaning is not fixed by the meaning of the individual words, and their combination. Human beings appear to handle figurative language effortlessly. These are properties that are hard to reproduce in machines.

The field of psycholinguistics investigates how humans process language. Psycholinguistics has amassed a considerable amount of evidence, and it has achieved insights into how humans process language that is of use in NLP. Jurafsky and Martin (2009, p. 48) note that the field would do well to borrow designs from nature, i.e. humans.

The related field of computational linguistics (CL) seeks to investigate linguistic hypotheses computationally. The connection to psycholinguistics should be more direct in CL than in NLP, as linguistic hypotheses are indeed about humans. King (1992) pointed out a lack of interaction between psycholinguistic evidence and CL when designing programs that process the meaning of sentences. She noted that few computational linguists have used psycholinguistic findings to inform their design choices, and when it was done, it was not very comprehensive.

The field of psycholinguistics is a far way from having a fully specified account of how language understanding proceeds. However, in terms of the general architechture of the process, findings are more definite. The present text is about developing a psycholinguistically plausible architechture for the task of language understanding.

1.2 Goals

The main goal of the present text is to construct a mathematical sketch of the achitechture of computer programs that process natural language in a way that mirrors the architechture of how humans process language according to psycholinguistic findings. Within the set of processes / programs that have structure conformant with this mathematical sketch will be a process that actually solves natural language understanding in a robust way. This is the question of "what is the equivalent of the standard pipeline in humans?". This follows if the mathematical sketch is made in accordance with findings from psycholinguistics, and these findings are themselves correct. The processes excluded from this set *might* work, but are not *supported* by psycholinguistic research.

A secondary goal in the present text is to investigate how a psycholinguistically inspired architechture for processing language can deal with the problems traditionally faced in natural language processing. These include computationally tractable scaling to multiple domains, and processing of figurative language. That is, I want to investigate the use of a psycholinguistically plausible architechture for important unsolved problems in natural language understanding. My secondary goal in this project is not to make the programs that solve important NLP problems, but describe the structure of such a program.

1.3 Methods and challenges

In order to make a mathematical sketch that mirrors results in psycholinguistics, I found it necessary to review important findings is psycholinguistics extensively. I have followed a bottom-up principle, attempting to get an overview of what results exist before making design choices. Psycholinguistic research is however, as most branches of psychology, tremendously fragmented into research concerning distinct phenomena. I have tried to summarize important findings on human language comprehension across research on particular psycholinguistic phenomena to provide a coherent picture of what is currently thought to be the case.

The present project concerns structural properties of processes, such as whether the process is incremental or processing proceeds in sentence or clause level chunks. This focus stands in contrast to a focus on implementation details. Compared to implementation details, structural properties of the process of language comprehension are low hanging fruit in psychology. Implementation details have proven very hard to get at, and can perhaps only be investigated with emerging advanced neural imaging techniques. Interestingly, psycholinguistic research, for instance on parsing, has often focused on highly specific implementation details. Rarely are the, in my opinion, much more solid findings on structural properties of processes taken stock of. This tendency adds an additional layer of complexity.

I chose to encode psycholinguistically plausible processes using coalgebras. Coalgebras are ordinarily used in theoretical computer science, and have found use in modelling the semantics of programming languages. Although a bit of abstract mathematical overhead is necessary in order to arrive at coalgebras, they are actually quite simple, and provide a formal framework for the representations of structural characteristics of processes. Coalgebras provide an intermediary between informal notions of processing found in psychology, and computer implementations. As mentioned, the general methodology is to first review and discuss findings in psycholinguistics, before translating the results into a coalgebraic representation. The model will be constructed in increments, beginning with elementary findings on parsing, and extending with additional structure. As the model is extended, research concerning possible applications to NLP problems is discussed.

1.4 Overview of the text

The next chapter (2) discusses the standard pipeline of natural language processing. The standard pipeline is the model to be replaced, and so it is important to discuss both its origin and the problems associated with it.

Chapter 3 introduces coalgebras, which are used for encoding the model. In chapter 4, I introduce basic findings on syntactic parsing and introduce the basic architechture for the model. Then, I review findings on reanalysis, and introduce a naïve algorithm based on these findings.

Chapter 5 introduces meaning representations to the model. I discuss how the behavior of the model extended with such representations relates to the model of syntactic parsing. First, I discuss psycholinguistic findings on this relationship, and then represent it formally. The reanalysis algorithm is then updated and related to the previous algorithm.

Chapters 6 and 7 discuss psycholinguistically motivated extensions of the model that may yield benefits in terms of computational complexity and the processing of some types of figurative language. Both these extensions introduce various types of context at an early stage of processing. This contrasts to the standard pipeline, where such processing happens later.

Chapter 6 extends the model with states of knowledge activation. I argue that such states are of use primarily in resolving word senses at a very early stage. Additionally, there is some evidence that they can influence syntactic decisions, improving the accuracy of the parser.

Chapter 7 extends the model with context models. These can help make semantic constraints on interpretation more dynamic, by overriding conceptual knowledge.

Chapter 2

The standard pipeline of natural language processing

The present chapter introduces the standard pipeline (2.1), as well as its origins (2.2). These sections motivate the usefulness of an architechture like the standard pipeline in natural language processing. There are however some problems with the standard pipeline, both theoretical (2.3.2) and practical 2.4. In upcoming chapters, I will discuss how the model proposed in this project can make headway on these problems.

2.1 The standard pipeline

The standard pipeline of NLP is a high level model for processing language from language input to meaning representation. The model specifies what the different processes in NLP do (in terms of input and output data types), and how these processes are structured in the task of natural language understanding. However, the standard pipeline often figures as a background assumption in NLP literature. It is often implicitly used to guide the way components are assembled to understand language, but seldom made entirely explicit.

An exception is Bird, Klein, and Loper (2009). They give an account of what they call “the commonly assumed pipeline for NLP” (p. 31).

> Simple pipeline architecture for a spoken dialogue system: Spoken input (top left) is analyzed, words are recognized, sentences are parsed and interpreted in context, ... different types of linguistic knowledge inform each stage of the process.

Although it is not made perfectly explicit, it appears that the standard pipeline is assumed to process sentence or clause level input in such successive stages. Both Bates (1995) and Allen (1995) give accounts of the standard pipeline that would suggest as much. Applications and models of natural language understanding with a pipelined approach also appear to understand it as operating on the sentence level (e.g. Miller, Stallard, Bobrow, & Schwartz, 1996; Novichkova, Egorov, & Daraselia, 2003).

I will assume that the standard pipeline is the view that language processing proceeds through successive stages of analysis of *sentence level input*. Note that the standard pipeline permits specialized processing of different types of information to happen in parallel. The results of two such processes is joined at a later stage. For instance, the word sense disambiguation and syntactic analysis occur in isolated, parallel pipes, their respective outputs joined as inputs to semantic processing.

2.2 Origins

2.2.1 Psychological origins

In *The Modularity of Mind*, J. A. Fodor described the modularity thesis, stating that input systems of the mind (most notably perception and language comprehension) are made up of modules that deal with specific types of information. Among the criteria listed for such modules, is the notion of informational encapsulation (J. A. Fodor, 1983, p. 64). That is, mental modules are postulated to operate in an independent way, impervious to outside influence once domain specific input data is specified. Language comprehension is one of these modules according to Fodor. If language comprehension itself consists of modules in the Fodorean sense, these should also be informationally encapsulated, leading naturally to a view of language comprehension as a pipeline.

Although inspired by psychological hypotheses, the standard pipeline is not a theory about how the mind works. It is a model of one possible way of assembling certain components in order to produce a mechanism that solves a task. We do not expect the standard pipeline to comport with human language processing. Once it is assembled however, it should be able to handle the task in a robust and comprehensive way.

2.2.2 A possible way of solving the sparse data problem

The sparse data problem is a problem faced when approximating a function, where the input-output pairs that exist in the training data only cover a small portion of the totality of inputs on which the function is defined. In the present situation, the function takes as input sentences of natural language (in context) and returns meaning representations.

The standard pipeline can be viewed as a strategy for acheiving comprehensive coverage of a function from sequences of words to meaning representations by subdividing the problem into a cascade of functions that process different types of information.

For instance, van den Bosch and Buchholz (2002) note that “In many early parsers, the POS sequences formed the only input to the parser, i.e. the actual words were not used except in POS tagging.” (p. 1). Simplifying a bit, these parsers would operate on sequences such as `Det Noun Verb Adj` instead of `The cat is fat`. van den Bosch and Buchholz (2002) further note that: «the main reason for the use of POS tags in parsing is that they provide useful generalizations and (thereby) counteract the sparse data problem.» (p. 2). Mathematically speaking, such parsers were defined on the quotient induced on the set of all sentences by mapping each sentence to its representation as a sequence of part of speech tags.

2.2.3 A way of dividing labor when engineering NLP systems

The standard pipeline can also be viewed as simply a way of dividing labor when constructing NLP systems (A. van den Bosch, personal communication, December 24, 2012). This is a common strategy in engineering. Teams of engineers work on different components with well specified inputs and outputs.

Ordinarily, when constructing a complex mechanism, it is clear beforehand that a system will indeed work when we assemble the components. The situation is not quite the same in natural language processing. It may well be that it is not feasible to make a robust mechanism that handles natural language using the components and architechure suggested by the standard pipeline. Again, it is clearly a prediction of the standard pipeline that the components will be able to solve the larger problem once assembled.

2.3 Theoretical problems associated with the standard pipeline

2.3.1 The standard pipeline lacks a basis in psycholinguistics

We do not have substantive evidence to suggest that the standard pipeline can indeed solve natural language understanding in a comprehensive way. Indeed, without any *psycholinguistic* evidence, it is unclear how we would know whether or not the standard pipeline can work short of actually trying it. The inadequate empirical foundation of the standard pipeline is the main area where the alternative model attempts to improve. More specific inadequacies of the standard pipeline exist, but these are only indirectly targeted by the alternative model. It is inaccurate to attribute these problems to a lacking correspondence with what is known about language comprehension in psycholinguistics. There may exist high level models of language processing that do not correspond to psycholinguistic results, and yet solve language comprehension problems in a robust and comprehensive way. Just as there are many ways of making a functioning eye, there may be many ways of making a working process for natural language comprehension.

2.3.2 The standard pipeline is not represented with a mathematical language

A second theoretical problem of the standard pipeline is that it not represented with a mathematical language. Rather, the standard pipeline is typically represented with a diagram. It is quite clear from a diagrammatic representation how the corresponding computer programs should be assembled. However, the fact that the standard pipeline is not represented by a mathematical language is problematic for the field of natural language processing. Having a language for representing high level, structural properties of processes would mean that the space of possible architechtures of NLP systems can be explored and analyzed in a precise and systematic way. At present, such a language appears to be lacking.

I employ the language of coalgebras to represent a psycholinguistically inspired model of language comprehension. I argue in chapter 3 that coalgebras are suitable for this purpose. Formalizing a high level architechture for processing language in coalgebras may in itself be useful, if important issues in the architechture of computer programs that process natural language are made precise.

2.4 Practical problems associated with the standard pipeline

Traditional approaches have attempted to solve various tasks in natural language processing with rule- and logic-based approaches that are structured in a way conformant with the standard pipeline. Manning and Schütze (2001) note that

> ... experience with AI approaches to parsing and disambiguation, which seek models with deep understanding, has shown that hand-coded syntactic constraints and preference rules are time consuming to build, do not scale up well, and are brittle in the extensive use of metaphor in language (Lakoff, 1987). (p. 18)

In order to work properly, these systems are often restricted to dealing with linguistic input in a single domain (Bates, 1995). Of course, there may be different causes for the failings of such systems other than their use of the standard pipeline.

The extension to the model discussed in chapter 7 is likely to have use in reducing the brittleness of selectional restrictions (i.e. preference rules) with respect to figurative language.

2.4.1 Combining highly accurate components does not yield a highly accurate mechanism

Symbolic variants of the standard pipeline are not forgiving of mistakes. Generally speaking, the standard pipeline offers no mechanism to recover from errors other than the outright failure of later stages of processing. Erroneous output is simply passed on to an oblivious next stage of processing. There, the error is amplified, as the subsequent stage will make processing decisions based on the error. By the very design of the standard pipeline, a minor, early error may lead to a very strange semantic reading of a sentence.

2.4.2 Computational complexity problems

In a *symbolic* pipelined NLP program, a stage of analysis will typically not lead to a unique refined structure, but to many guesses at refined structures. In general, the standard pipeline may lead to a situation where the information that is useful or necessary when resolving an ambiguity might not be available in the stage of processing in which it arises. Knowledge may be necessary for resolving some syntactic ambiguities, but not come into play until later stages of processing. Forcing a guess would simply result in errors, meaning both analyses have to be kept. As more decisions must be postponed until a later stage, the number of alternate structures becomes very large.

> Symbolic language processing is highly nondeterministic and often delivers large numbers of alternative results because it has no means of resolving the ambiguities that characterize ordinary language. This is for the clear and obvious reason that the resolution of ambiguities is not a linguistic matter. After a responsible job has been done of linguistic analysis, what remain are questions about the world. (Kay, 2005, p. 437)

It is generally recognized that natural language understanding requires that comprehension processes somehow interface with knowledge. An interface with knowledge is possible in the standard pipeline in a cumbersome way, through backtracking or inferencing once sentence level semantic representations are produced. This troublesome relationship may at least partly explain the high branching factor associated with the standard pipeline. It is worth noting that such a relationship should be dynamic, to cover metaphorical phenomena that are not strictly in keeping with literal knowledge.

As mentioned earlier, Manning and Schütze (2001) argue that symbolic NLP systems (many of which use the standard pipeline) do not scale well. For instance, expanding grammatical coverage generally means more syntactic decisions will need to be made for any sentence. If these are explored in parallel, computational complexity becomes a big problem (p. 18). Similarly, extending the lexicon can translate into many more syntactic ambiguities. The same tokens now may admit more part-of-speech tags, leading sentences to admit many more syntactic analyses. Extending the lexicon also means that there are more possible meanings that can be assigned to words, and many more possible meanings that could be assigned to sentences.

The extension in chapter 5 will likely be of use in disregarding impossible syntactic analyses at an early stage, while extensions in chapter 7 will nuance decisions on which analyses should be kept and disregarded depending on context.

The extension to the model proposed in chapter 6 is likely of benefit in reducing computational complexity by resolving many semantic ambiguities before they multiply, and by improving the accuracy of initial parsing decisions, thus reducing the need for costly reprocessing.

Chapter 3

Mathematics

The following chapter introduces the mathematical formalisms used to encode the model. The coalgebra formalism, (as employed here) allows the encoding of distinctions regarding the general shape of a process, at approximately the same level of abstraction as the standard pipeline.

3.1 Motivation in brief

Mathematical functions are often presented as black box abstractions over processes in introductions to discrete mathematics. That is, a function can be seen as an abstraction over a deterministic process taking inputs from a range, and providing outputs in a domain. Functions abstract away how the association is accomplished, and simply store the input-output associations. Coalgebras on sets can be thought of as a formal and systematic way of constructing special functions that correspond by way of the black box analogy to processes with different properties. Coalgebras thus establish a diverse array of types of processes. From this array, one can choose a distinguished candidate that corresponds to psycholinguistic findings.

The first of the following sections 3.2 introduces commonplace definitions in category theory. These definitions, although we will see little of them in the remainder of the text, form the formal foundations of coalgebras. The second of the following sections (3.3) introduces the basic endofunctors on the category of sets. As discussed by Rutten (2000), these functors are important in defining high-level properties of processes, such as the involvement of output and termination. The duality with algebra is briefly discussed in 3.4. In 3.5, further functors are introduced that are involved in defining other properties of processes, such as input and determinism. The motivation behind coalgebras as a suitable formalism for encoding distinctions regarding the shape of a process is elaborated in section 3.6.

3.2 Basic category theory

The presentation of basic category theory contained here follows presentations in Wolter (2012) and Barr and Wells (1999). The main point of the present subsection is to introduce functors, specifically functors on the category of sets, as coalgebras are defined on these.

A common way (cf. Rosen, 2007, p. 591) of defining a graph G is as a pair (V, E), where V is a nonempty set of vertices and E is a set of edges, where the edges (for directed graphs) can be represented by a relation on V ($E \subseteq V \times V$), with each $(v_1, v_2) \in E$ representing an edge from the vertice v_1 to the vertice v_2. Essentially, directed graphs represent edges by their source and target vertices, meaning that there can be maximally one edge from a vertex v_1 to a vertex v_2. The following definition extends the above definition by adding identities for each edge, allowing there to be multiple edges between two vertices.

Definition 3.2.1 (Graph)**.**
Following the definition in Wolter (2012, p. 9), a graph is a quadruple (G_0, G_1, src^G, trg^G) where
G_0 is a collection of vertices,
G_1 is a collection of edges,
$src^G : G_0 \to G_1$ is a function assigning each edge e in G_1 its source vertice $src(e)$.
$trg^G : G_0 \to G_1$ is a function assigning each edge e in E to its target vertice $trg(e)$.
G_0, G_1 are collections to allow for the case where they are classes and not sets. A vertice or an edge will sometimes be said to be in the graph G, when technically speaking these are in G_0 and G_1 respectively.

Definition 3.2.2 (Graph Homomorphism)**.**
Barr and Wells (1999) give the following standard definition of a graph homomorphism (p. 11). Let $G = (G_0, G_1, src^G, trg^G), H = (H_0, H_1, src^H, trg^H)$ be graphs. A graph homomorphism $f : G \to H$ is a pair of functions $f = (f_0, f_1)$ mapping vertices to vertices and edges to edges from one graph to the other.:
$f_0 : G_0 \to H_0$,
$f_1 : G_1 \to H_1$

Satisfying the following structure-preserving criterion:
For all e in G_1:
$f_V(src^G(e)) = src^H(f_1(e))$
$f_V(trg^G(e)) = trg^H(f_1(e))$

Stating essentially that positive statements about the targets and sources of edges in G are still true about their respective counterparts in H (Lawvere & Schanuel, 2009, Session 14). For instance, we see that for all e, e' in G_1 with $src^G(e) = trg^G(e')$, we also have $src^H(f_1(e)) = trg^H(f_1(e'))$. That is, if the source of e is the same as the target of e' (in G_0), the source of

$f_1(e)$ will be the same as the source of $f_1(e')$. It is however possible for instance that the target of e is different from the target of e' in G_0, but for these to be identified for their respective counterparts in H.

Example 3.2.1 (Graph Homomorphism)**.**
Consider the graphs G and H given below.

Figure 3.1: Graph G

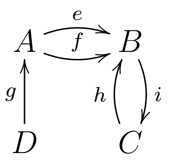
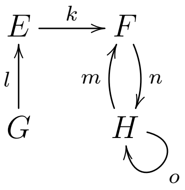
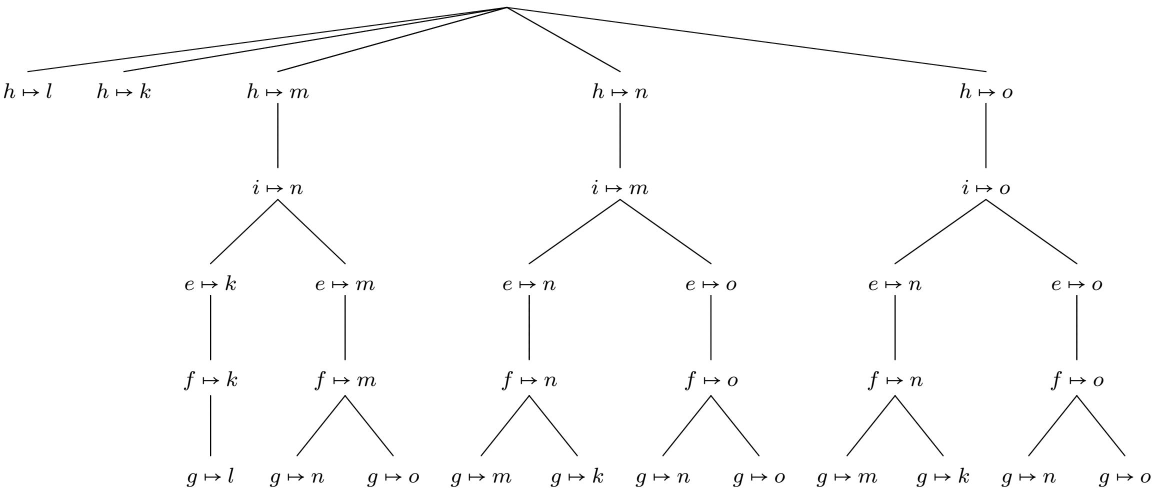


Figure 3.2: Graph H



From G to H the following homomorphisms exist:

Figure 3.3: Graph homomorphisms from G to H



Homomorphisms are denoted by decending paths starting in an assignment of h, ending with an assignment of g. Note that since the graph G is (weakly) connected, the behavior of a graph homomorphism on vertices is forced by the behavior on edges by the structure-preserving criterion on graph homomorphisms. The homomorphisms from G to H can be seen as a subset of the pairs of functions $\phi_i : H_i \to G_i, i = 0, 1$. There are no homomorphisms from H to G as there is no edge a preserving the condition $src(a) = trg(a)$ required by $o \mapsto a$, i.e. there is no looping edge in G.

Definition 3.2.3 (Category)**.**
The definition of a category given here follows in broad strokes the definition in Wolter (2012, p. 14). A category $\mathbb{C}$ is given by:

- A graph $C = (C_0, C_1, src^C, trg^C)$

- A partial composition operation $\circ^{\mathbb{C}} : C_1 \times C_1 \rightharpoonup C_1$ with $f \circ^{\mathbb{C}} g$ defined for f, g in C_1 with $src^C(f) = trg^C(g)$.
- A function $id_A : C_0 \to C_1$, assigning to each node A their identity edge $id_A^{\mathbb{C}}$.

Members of C_0 (nodes) are denoted the *objects* of the category, while members in C_1 are the *morphisms* relating the objects. The superscript $^{\mathbb{C}}$ is omitted whenever it is clear from context in which category the composition is taking place. Any category $\mathbb{C}$ is subject to the following properties.

- Identity
  For all objects A in $\mathbb{C}$, there exists a morphism id_A with $sc(id_A) = tg(id_A) = A$ such that:
  for all morphisms f in $\mathbb{C}$ with $tg(f) = A$: $id_A \circ f = f$
  for all morphisms g in $\mathbb{C}$ with $sc(g) = A$: $g \circ id_A = g$.
- Associativity
  For for all morphisms f, g, h in $\mathbb{C}$ with $sc(g) = tg(h), sc(f) = tg(g)$:
  $f \circ (g \circ h) = (f \circ g) \circ h$.
- Book-keeping
  If $f \circ g$ is defined for f, g in $\mathbb{C}$, we have $sc(f \circ g) = sc(g)$ and $tg(f \circ g) = tg(f)$.
  Additionally for all objects A, $src(id_A) = trg(id_A) = A$.

Definition 3.2.4 (Category of sets)**.**
The following definition is from Wolter (2012, p. 10) The category of sets is given by:
Objects: The collection of all sets
Morphisms: The collection of all (total) functions between them.
Composition is given in the usual way for functions. The identity on an arbitrary set A is given by the function id_A^{Set} given for all $a \in A$ by $id_A(a) := a$.

Definition 3.2.5 (Functor)**.**
The following definition is from Barr and Wells (1999, p. 65). Let $\mathbb{C}, \mathbb{D}$ be categories. A functor $F : \mathbb{C} \to \mathbb{D}$ is given by a graph homomorphism $F : C \to D$, where C and D are the underlying graphs of categories $\mathbb{C}$ and $\mathbb{D}$ respectively. Furthermore, F respects the category-structure of $\mathbb{C}$ and $\mathbb{D}$ by conforming to the following requirements:

- Identities are preserved:
  For all objects A in $\mathbb{C}$: $F(id_A^{\mathbb{C}}) = id_{F_0(A)}^{\mathbb{D}}$
- Composition is preserved:
  For all objects A, B, E, morphisms $g : A \to B$, $f : B \to C$ in $\mathbb{C}$: $F_1(f \circ^{\mathbb{C}} g) = F_1(f) \circ^{\mathbb{D}} F_1(g)$

That is, identities in $\mathbb{C}$ are sent to the corresponding identities in $\mathbb{D}$, and it does not matter if we compose two morphisms in $\mathbb{C}$ and then translate the result to a morphism in $\mathbb{D}$, or if the morphisms are first translated to morphisms in $\mathbb{D}$, and only then composed (in $\mathbb{D}$). Also note that the property that composable morphisms in $\mathbb{C}$ are also composable in $\mathbb{D}$ is inherited from the fact that F is a graph homomorphism. When $\mathbb{C} = \mathbb{D}$, we say that F is an *endofunctor*.

3.3 Some endofunctors on sets and their coalgebras

The purpose of this section is to introduce coalgebras on sets. The presentiation is inspired by far richer presentations in Jacobs and Rutten (2011) and Rutten (2000). As mentioned, I will encode the model using coalgebras. First, I will introduce the notion of a coalgebra. As various endofunctors on the category of sets are defined, I will describe example coalgebras using such functors in order to elucidate the behavioral properties that are represented by the functor. Definitions of the endofunctors mentioned in this chapter can be found in Jacobs (2012, p.)

Definition 3.3.1 (Coalgebra)**.**
Rutten (2000, p.8) defines a coalgebra in the following way. Let $F : Set \to Set$ be a functor from the category of sets to the category of sets (i.e. an *endofunctor* on sets)[1]. An F-coalgebra is given by a tuple (S, c) where S is a set and $c : S \to F(S)$ is a function. S is referred to as the *carrier* of the coalgebra. Its members can be thought of as states. The function c describes for each state $s \in S$ a behavior.

Definition 3.3.2 (Identity functor on sets)**.**
The identity functor on sets, has behavior on objects (sets) X given by:

$$X \mapsto X$$

Behavior on morphisms (functions) $f : X \to Y$ is given by:

$$f \mapsto f$$

The identity functor obviously both preserves identities and composition, as both requirements follow by the reflexivity of = on morphisms. The identity functor will be identified by how it behaves on objects, with behavior on morphisms as above.

Example 3.3.1 (Identity functor coalgebras)**.**
Define a set S of states: $S := \{s_1, s_2\}$ A identity-functor coalgebra with carrier S is a tuple (S, c) where c is a function:

$$c : S \mapsto S$$

---

[1] Technically speaking, F need not be an endofunctor on the category of sets, but only these coalgebras are considered in the present text.

For instance, we might define c by $s_1 \mapsto s_2, s_2 \mapsto s_1$. The behavior of the coalgebra is then very simple, in state s_1, c transitions to state s_2, in state s_2, switch to s_1.

More generally, the identity functor coalgebras are the state transition systems where the transition is a function.

Definition 3.3.3 (Constant functor on Sets)**.**
Let A be a set. The constant functor on sets, has behavior on objects given by:

$$X \mapsto A$$

Behavior on morphisms $f : X \to Y$ is given by

$$f \mapsto id_A$$

The identity condition is satisfied, as the identity on any set is also assigned to id_A. The compositionality condition is satisfied as for any two composable morphisms $f : A \to B, g : B \to C$ the morphisms are both assigned to id_A, additionally, their composition $g \circ f$ is assigned to id_A. By the identity condition on categories, we have $id_A \circ id_A = id_A$.

Note that in the notation specifying functors, X is free to vary over sets, while capital letters are not X, e.g. A are taken as arbitrary but fixed for the given functor.

Example 3.3.2 (A coalgebra on a constant functor)**.**
As the constant A in a constant functor assigning an arbitrary set X to A is fixed, all coalgebras on such a constant functor must be functions with codomain A.

For instance, consider a set of states $S := \{s_1, s_2, s_3\}, A := \{red, green\}$. Define a constant functor-coalgebra (where A is the constant) (S, c) where $c : S \to A$ is given by:

$$s_1 \mapsto red, s_2 \mapsto red, s_3 \mapsto green$$

In other words, coalgebras on the constant functor associate states with values in A, which in the present case are colors. No transitions are associated with states.

How could one define a functor where the coalgebras associate both states and and values with states? First, the product functor has to be introduced.

Definition 3.3.4 (Product functor on Sets)**.**
Let F, G endofunctors on Sets. The corresponding product functor on Sets, has behavior on objects given by:

$$X \mapsto F(X) \times G(X)$$

Behavior on morphisms $f : X \to Y$ is given by:

$$f \mapsto F(f) \times G(f)$$

Where $F(f) \times G(f) : F(X) \times G(X) \to F(Y) \times G(Y)$ is given for all $(z, v) \in F(X) \times F(Y)$

$$F(f) \times G(f)(z, v) := (F(f)(z), G(f)(v))$$

Example 3.3.3 (A product coalgebra with values and transitions)**.**
Assume the special case of the product functor on sets where F and G are the constant functor assigning X to A and the identity functor respectively. F given below is the corresponding product functor.

$$X \mapsto A \times X$$

Let $A := \{red, green\}$ and $S := \{s_1, s_2, s_3\}$ as before. Define a coalgbra (S, c), where $c : S \to A \times S$ is given by:

$$s_1 \mapsto (red, s_1), s_2 \mapsto (red, s_3), s_3 \mapsto (green, s_2)$$

The coalgebra associates with states both a value / label and a new state. The fact that states are assigned values is due to the involvement of the constant functor. The fact that states are assigned to new states is due to the involvement of the identity functor. The fact that both values and new states are assigned to states is due to the corresponding product functor (Rutten, 2000, Section 3).

A different situation is one where states are *either* associated with a new state or with a symbol. In order to represent such mechanisms coalgebraically, the sum functor must be introduced.

Definition 3.3.5 (Sum functor on Sets)**.**
For sets A, B we define their disjoint union $A \uplus B$ in the following way:

$$A \uplus B = \{(a, 1) : a \in A\} \cup \{(b, 2) : b \in B\}$$

This construction is also denoted the sum $A + B$ in Set. A heuristic for remembering this fact is that $|A| + |B| = |A \uplus B|$ for all sets A, B, as every member of A, B has a unique representative in $A \uplus B$ (and every member of $A \uplus B$ represents some element of A or of B). Elements of $A \uplus B$ will often be referred to without indices whenever it is clear from context where they originate.

Let F, G be endofunctors on Sets. The sum of their respective behaviors on objects (sets) can be extended to a functor given by:

$$X \mapsto F(X) + G(X)$$

With behavior on morphisms $f : X \to Y$:

$$f \mapsto F(f) + G(f)$$

Where $F(f) + G(f) : F(X) \to G(X)$ is given for $(z, i) \in F(X) + G(X)$ by the assignment:

$$F(f) + G(f)(z, i) := [F(f), G(f)] = \begin{cases} (F(f), i) & \text{if } i = 1, \\ (G(f), i) & \text{if } i = 2 \end{cases}$$

Example 3.3.4 (A coalgebra on a simple sum functor)**.**
Let F and G in the definition of the sum functor be the constant functor $X \mapsto A$ and the identity functor respectively. The corresponding sum functor is then:

$$X \mapsto A + X$$

Let $A := \{red\}$ and $S := \{s_1, s_2, s_3\}$. Define coalgebra on the sum functor above (S, c) given by:

$$s_1 \mapsto s_1, s_2 \mapsto s_3, s_3 \mapsto red$$

This coalgebra has the following behavior. In state s_1, we are for ever stuck in s_1. In state s_2, we change state to s_3, which leads to red, and no new state. The presence of the sum functor typically means there is a choice between two types of behavior. Either we get a value from A and no new state, or we get a new state but no value in A. The A+ part of the functor essentially ensures there is a way of terminating (Rutten, 2000, p. 15).

Definition 3.3.6 (Set-indexed sum functor)**.** The set-indexed sum functor is a more general sum functor. Let $(F_i)_{i \in I}$ be a family of endofunctors on the category of sets, and I be any set. Let A be an arbitrary but fixed set.

$$\coprod_{i \in I} (F_i)(A) := \bigcup_{i \in I} (\{(x, i) : x \in F_i(A)\})$$

The sum of their respective behaviors on objects (sets) can be extended to a functor $\coprod_{i \in I}(F_i)$ given by:

$$X \mapsto \coprod_{i \in I} (F_i)(X)$$

With behavior on morphisms $f : X \to Y$:

$$f \mapsto \coprod_{i \in I} (F_i(f))$$

Where $\coprod_{i \in I}(F_i(f)) : \coprod_{i \in I}(F_i(X)) \to \coprod_{i \in I}(F_i(Y))$ is given for $(z, i) \in \coprod_{i \in I}(F_i(X))$ by the assignment:

$$\coprod_{i \in I} (F_i(f))(z, i) := (F_i(f)(z), i)$$

In the next sections, as a part of a discussion on the duality of coalgebras with algebras, I discuss an additional example of a coalgebra on a functor constructed using constant, identity, sum and product functors.

3.4 Duality with algebra

The duality of coalgebras with algebras can further elucidate the notion of a coalgebra. First, a definition of an algebra is in order.

Definition 3.4.1 (Algebra)**.**
Let $F : Set \to Set$ be an endofunctor on the category of sets. An F-algebra is given by a tuple (A,a) where A is a set and $a : F(A) \to A$ is a function (Jacobs & Rutten, 2011, p. 16).

The name *co*algebra stems from the fact that algebras can be encoded by functions $g : F(C) \to C$. Coalgebras represent the opposite notion, as functions the other way are considered. According to Jacobs and Rutten (2011), this opposition is reflected in the phenomena captured by algebras and coalgebras: "The basic dichotomy may be described as *construction* versus *observation*" (p. 5, emphasis original). In the example below, the algebra describes how to construct new objects from existing objects and elements of A. A coalgebra however, describes how to obtain A and a new object from an existing object. Here, the objects are better thought of as states.

Example 3.4.1 (A simple algebra)**.**
In algebra, elements of the carrier are often thought of as structures. Consider the functor G given by the assignment:

$$X \mapsto 1 + (A \times X)$$

Where $1 := \{*\}$, a singleton set. If we take as carrier A^*, we can define an algebra (Jacobs & Rutten, 2011, p. 18):

$$[empty, cons] : G(A^*) = 1 + (A \times A^*) \to A^*$$

$[empty, cons]$ denotes the function resulting from applying $empty$ to members in $1 + (A \times A^*)$ originating in 1, $cons$ to those originating in $A \times A^*$. Under the ordinary interpretations of A^* and $cons$, each $l \in A^*$ is reached by repeated applications of $cons$, starting from $empty(*) = []$. That is, the algebra describes how members of the carrier set are built.

Example 3.4.2 (A simple coalgebra)**.**
Let b be a G-coalgebra, with S, understood as a set of states, as carrier.

$$b : S \to 1 + (A \times S)$$

In general, each s in S will be associated with either a pair (σ, s') where $\sigma \in A$ and $s' \in S$, or with an element of 1, in which case there is no new $s' \in S$. The first coordinate σ can be thought of as an emitted behavior, whereas $* \in 1$ may be thought of as termination. That is, s is associated with a finite or infinite sequence of elements (emitted behavior) of A (Rutten, 2000,

p. 18). The finite sequences are those that eventually lead to termination. Thus, the coalgebra b encodes the future behavior / future observations in the states of S.

The functor on which an algebra is defines what kind of construction processes are available to build new objects. As such, they characterize different types of objects in a very abstract way. Dually, the functor on which a coalgebra is defined, constrains the behaviors of the coalgebra. As such, functors characterize different types of behaviors in a very abstract way.

3.5 Further functors

Definition 3.5.1 (Exponent functor)**.**
Let F be an endofunctor on Sets, and A and arbitrary but fixed set. The corresponding exponent functor is given by the following assignments.

$$X \mapsto F(X)^A$$

On a morphism $f : X \to Y$, the exponent functor behaves in the following way:

$$f \mapsto F(f)^A$$

Where $F(f)^{\cdot}F(X)^A \to F(Y)^A$ is given for all $g : A \to X$

$$F(f)^A(g : A \to X) := F(f) \circ g$$

Example 3.5.1 (Coalgebras on two exponent functors)**.**
The exponent functor encodes the notion that behavior of a mechanism is dependent on input Rutten (2000, p.16). However, the way the functor is built describes what behaviors should be dependent on input in the coalgebras. Consider a coalgebras of the functor:

$$X \mapsto (A \times X)^B$$

Where $S := \{s_1, s_2\}$, $A := \{red, green\}$, $B := \{switch, stay\}$. Let (S, c) be given by

$$\begin{aligned} c(s_1)(switch) &:= (red, s_2) \\ c(s_1)(stay) &:= (green, s_1) \\ c(s_2)(switch) &:= (green, s_1) \\ c(s_2)(stay) &:= (red, s_2) \end{aligned}$$

In this case, choice of color is dependent on both state and input. Consider however a coalgebra (S, d) on the functor below:

$$X \mapsto A \times (X^B)$$

Where d is given by:

$$\begin{aligned} d(s_1) &:= (red, [switch \mapsto s_2, stay \mapsto s_1]) \\ d(s_2) &:= (green, [switch \mapsto s_1, stay \mapsto s_2]) \end{aligned}$$

In this case, the value in A does not depend on a value of B, but is given once we know the identity of a state of the carrier.

Definition 3.5.2 (Power set functor**).**
Let F be an endofunctor on Sets. Define the power set functor $\wp(F)$ by the assignments:

$$X \mapsto \wp(F(X))$$

On a morphism $f : X \to Y$, the exponent functor behaves in the following way:

$$f \mapsto \wp(F(f))$$

Where the function $\wp(F(f)) : \wp(F(X)) \to \wp(F(Y))$ is given for all $Z \subseteq F(X)$ by its image under $F(f)$, i.e.

$$\wp(F(f))(Z) := \{y : \exists z(z \in Z \wedge F(f)(z) = y)\}$$

Example 3.5.2 (Finite State transducer**).**
A finite state transducer (FST) can be encoded as a coalgebra. The example is similar to the example of a coalgebraic representation of deterministic finite state automata given by Rutten (2000, p. 16). Following Roche and Schabes (1997, p. 14), a FST is defined in the following way:

- Q is a finite set of states
- Σ and Δ are sets of input and output symbols respectively
- $q \in Q$ is the initial state
- $F \subseteq Q$ are a set of final states.
- A relation $E \subseteq Q \times \Sigma^* \times \Delta^* \times Q$ describing output and transitions.

The output and transition function can be captured by a coalgebra. Define a functor $G : Set \to Set$ by the assignment:

$$X \mapsto (\wp(\Delta^* \times X))^{\Sigma^*}$$

The involvement of the product with a fixed set encodes the involvement of input, the involvement of the powerset-functor encodes the fact that transitions are nondeterministic. Finally, the involvement of the exponent functor with a fixed exponent encodes the fact that transitions are dependent on input. Fixing X to a set Q, a G-coalgebra t (below) encodes exactly one relation E, and any relation such as E is encoded by exactly one G-coalgebra such as t with carrier Q.

$$t : Q \to (\wp(\Delta^* \times Q))^{\Sigma^*}$$

That is, once a set of states Q is fixed, there is a bijective correspondence between the set of relations such as E describing transitions and output above and the G-coalgebras with Q as carrier. For all sets Q, we have:
$\wp(Q \times \Sigma^* \times \Delta^* \times Q) \cong ((\wp(\Delta^* \times Q))^{\Sigma^*})^Q$.

Proof.
For all sets A, B, C we are guaranteed isomorphisms such that:

$$\wp(A \times B) \cong (\wp(B))^A$$
$$A^{B \times C} \cong (A^B)^C$$

Applying these isomorphisms, the following result is obtained:

$$\begin{aligned}
&\wp(Q \times \Sigma^* \times \Delta^* \times Q) \\
&\cong (\wp(\Delta^* \times Q))^{Q \times \Sigma^*} \\
&\cong (\wp(\Delta^* \times Q))^{\Sigma^* \times Q} \\
&\cong ((\wp(\Delta^* \times Q))^{\Sigma^*})^Q
\end{aligned}$$

□

The Kripke polynomial functors are the functors that can be constructed from the endofunctors on set introduced thus far. These are the functors that will be considered in the present text. They are defined below.

Definition 3.5.3 (Kripke polynomial functor)**.**
The Kripke polynomial functors (KPF) on Sets are defined recursively by (Jacobs, 2012, pp. 36-37) In the basic case:

- The identity functor (3.3.2) is a KPF.
  $X \mapsto X$
- Additionally, if A is any set, the constant functor (3.3.3) is a KPF.
  $X \mapsto A$

Recursively:

- If $(F_i)_{i \in I}$ is a family of KPF endofunctors on sets,(I any set), the corresponding indexed sum functor (3.3.5) is a KPF.
  $X \mapsto \coprod_{i \in I} F_i(X)$

- If F,G are KPF, the corresponding product functor (3.3.4) is a KPF.
  $X \mapsto F(X) \times F(Y)$

- If F is a KPF, A is an arbitrary but fixed set, the corresponding exponent functor (3.5.1) is a KPF.
  $X \mapsto F(X)^A$

- If F is a KPF, the corresponding powerset functor (3.5.2) is a KPF.
  $X \mapsto \wp(F(X))$

3.6 Motivation for using coalgebras

Essential properties of state based mechanisms can be encoded coalgebraically by varying the Kripke polynomial functor on which the coalgebra is defined.

When we consider coalgebras as endofunctors on Set, they end up being functions with particular domains and ranges. As mentioned, in introductory texts about discrete mathematics, functions are often presented as black box abstractions over an underlying process. Coalgebra can be seen as extending and formalizing this view, by specifying explicitly characteristics of the black box. Following Rutten (2000, pp. 13-24), a statement that something is a coalgebra on some Kripke Polynomial Functor F with some carrier A can say something about:

- Are many new states (nondeterminism) or is a unique state (determinism) produced as output?

- Is there a way to fail to produce a new state?

- Are observations associated with transitions (what do these observations look like, i.e. what set do they come from)?

- Is input required to make transitions (and what does the input look like, i.e. member of which set?)

- Structural relations between these properties, such as failure being either dependent or independent of input.

Varying the functors does not give precise control over a mechanism, but constrains input, output and the architechture of the mechanism. Distinctions that provide a general picture of the shape of the process by which language is comprehended can however be encoded. An abstract formalism such as this is needed, as implementation details of human language comprehension are far from fully explored.

Another reason for using coalgebras is that allow the notion of likeness in behavior (but not implementation) to be defined elegantly. The notion of the *behavioral equivalence* processes is denoted a *bisimulation* between processes. This concept will turn out to be useful in representing the relation between parsers with and without meaning representations, as discussed in chapter 5.

Finally, coalgebras are used as semantics of programming languages. In fact, the coalgebra that will be used in the present text has similarities with the coalgebraic semantics of Java presented in Jacobs (1995). For instance, a class with two methods 1 and 2 may be thought of as a coalgebra on a functor:

$$X \mapsto (Errors_1 + (Out_1 \times X))^{In_1} + (Errors_2 + (Out_2 \times X))^{In_2}$$

According to Jacobs (1995), the carrier (states) of the coalgebra may be thought of as the instances of the class, and function as the class[2]. This means that formulating the model in terms of coalgebras is not far away from defining a computer program.

---

[2] The example above is simplified, the functors used in Jacobs (1995) are more elaborate.

Chapter 4

Evidence and modelling of syntactic processing

The present chapter focuses on basic evidence on syntactic processing, and how such evidence can be modelled coalgebraically. I review psycholinguistic findings on parsing (4.1), discuss the well-established finding that parsing is incremental is a word-by-word process (4.1.1). Moreover, I argue that this process does not explore syntactic alternatives in parallel in the same way that parallel algorithms from NLP do (4.1.2).

In section 4.2, I propose a preliminary architechture for incremental syntactic parsing. This preliminary architechture is then shown to be subsumed by the incremental parsing formalism of Nivre (2008). Next, I define a simple iterative procedure for processing sentences (4.2.4). The procedure is however inadequate, as it lacks completeness. In section 4.3, I discuss evidence on reanalysis, before I propose a reanalysis algorithm.

4.1 Incrementality and seriality in syntactic processing

In the present section, I will argue that syntactic processing has the following properties:

1. Syntactic parsing proceeds by incremental construction, on a word by word basis.

2. Syntactic structures are not explored in parallel.

3. There is facility for reranking or reanalysis upon failing to extend a chosen analysis.

These properties will be important in justifying design choices when I define the preliminary model in the next section. Moreover, these properties sidestep ongoing discussions about implementation details of human syntactic processing.

4.1.1 Syntactic processing is incremental and it is not unranked parallel

If parsing is incremental, it means that syntactic structures, however multiplicitous, are constructed incrementally as processing proceeds. Serial parsing is the notion that only one syntactic structure is constructed at a time. Following Pickering and R. P. G. van Gompel (2006), I will use the term *unranked parallel* to denote the idea that multiple syntactic represetations are incremented in parallel, without any preference or priority assigned to any of the representations.

If syntactic processing is incremental, comprehenders should be able to tell *while they are processing a sentence* that a syntactic hypothesis they have chosen is erroneous. Additionally, if syntactic processing is unprioritized parallel, no syntactic reading of a sentence should be given priority during comprehension. There should not be sentences where the resolution of a local ambiguity in one way yields processing difficulty, but the resolution of an ambiguity another way does not. Indeed such sentences exist. Sentences that lead comprehenders astray in their initial analysis are called *garden path* sentences.

The prototypical example of such a sentence from Bever (1970, p. 316) is (1) below.

(1) The horse raced past the barn fell.

Readers will typically adopt a reading where the horse is the subject of "raced", and encounter difficulty when reading the word "fell". No such difficulty obtains if the sentence ends with "quickly" instead, supporting the initial analysis adopted by the comprehender.

Frazier and Rayner (1982) used an eye-tracking paradigm to investigate processing when comprehenders read garden-path sentences. Among the types of sentences they investigated, were pairs of sentences such as the following (p. 184).

(2) a. Since Jay always jogs a mile and a half this seems like a short distance to him.
b. Since Jay always jogs a mile and a half seems like a very short distance to him.

Here, two syntactic analyses of both sentences are possible after "a mile and a half" is read. One where "a mile and a half" attaches to the verb phrase with "jogs" as head, and one where it does not. Subsequently the word "this" disambiguates sentence (2-a), whereas "seems" disambiguates (2-b). According to Frazier and Rayner, syntactically based principles (i.e. non-semantic) principles of minimal attachment and late closure govern how syntactic structures are incremented. Late closure here dictates that "a mile and a half" should attach to the verb phrase. Thus, comprehenders should spend more time reading the disambiguating word in (2-b), but not in (2-a). This is precisely what was found. Hence, the readers chose one analysis over another, as difficulty did not occur when the non-chosen analysis was ruled out. Moreover, they analyzed the sentence incrementally as they read from left to right, as difficulty occurred immediately after the chosen reading was disconfirmed.

As noted by Pickering and R. P. G. van Gompel (2006, p. 455), this finding has been replicated in many studies using the same paradigm. Syntactic processing is now viewed as incremental, whereas parallel approaches to parsing have been weakened to handle this finding. The next section discusses theories and evidence on serial vs. ranked parallel approaches to syntactic parsing.

4.1.2 In what sense are ranked parallel approaches to parsing parallel?

The question of parallelism in human parsing can be understood in two ways. The first of these concerns momentary parallelism as parsing decisions are made. The latter of these two senses is similar to parallel approaches to parsing in NLP.

1. Different syntactic structures are momentarily considered when deciding how to include an incoming word into the developing syntactic structure.
2. Different syntactic structures are elaborated (incremented) in parallel (i.e. breadth first).

I will argue that the mainstream approach to parsing that uses parallel syntactic representations are parallel in the first sense, and not in the second sense. Moreover, there is little reason to suppose parallelism of the second sense.

As mentioned in the last section, garden path phenomena mean sytactic theories that involve parallel syntactic representations must assume that there is a preference relation on such representations, with one structure being most preferred. Pickering and R. P. G. van Gompel (2006, p. 456) remarks that “All current accounts assume that syntactic processing is either serial or ranked parallel”. In ranked parallel models, processing difficulty on disambiguating words are thought to result from reranking of the preferred structure (Pickering & R. P. G. van Gompel, 2006, p. 456).

The approach of MacDonald, Pearlmutter, and Seidenberg (1994) an influential model of language comprehension making use of parallel representations cited as the canonical representative of such theories by Pickering and R. P. G. van Gompel. MacDonald et al. view sentence processing as a constraint satisfaction problem.

> ... the lexical representation for a word includes a representations of the word’s phonological form, orthographic form, semantics, grammatical features (including grammatical category), morphology (at least inflectional), argument structure, and X-bar structure. Words associated with more than one representation at one of these levels have all representations listed as alternatives. Comprehension involves computing a single alternative at each level when a word (ambiguous or not) is encountered. (p. 684)

Note however that the non-chosen representations are thought to linger and indeed be selected through re-ranking at later stages of processing in constraint based parsing.

Arguably, MacDonald et al. posit parallelism of the first kind, and in fact posit that parallelism of the second kind does not occur in language comprehension. That is, parallel representations are considered as the incremented state is decided upon, but multiple representations are not incremented. Alternative syntactic representations are not maintained in MacDonald et al. (1994). Indeed, there are good reasons to assume that they are not.

- If multiple analyses were maintained, there should be some processing difficulty associated when nonpreferred analyses are disconfirmed in locally ambigous sentences where readers are led to adopt the correct analysis (one could denote these the anti-garden path sentences).
- Moreover, in Frazier and Rayner (1982) and subsequent research utilizing the same paradigm, readers tend to regress when garden path sentences are disambiguated. Often, they will regress to the point where the ambiguity arose. If parallel representations are maintained, there should simply be no need to read the words anew.
- Parallel representations are not simply reranked when ambiguity is discovered, but the representation being ranked highest must be updated with the inputs (words) that were presented between the time was not adopted until it was ranked highest.

In summary then, it appears that the theory of MacDonald et al. (1994) considers parsing to be parallel in a different sense than what the parallel parsers we see in natural language processing on a computer (e.g. the CYK algorithm (cf. Jurafsky & Martin, 2009, p. 470)). The disagreement about parallel and serial processing concerns what happens in incremental stages of processing.

4.2 A preliminary architechture for syntactic parsing

For convenience, I repeat the summary of the previous section.

1. Syntactic parsing proceeds by incremental construction, on a word by word basis.
2. Syntactic structures are not explored in parallel.
3. There is facility for reanalysis upon failing to extend a chosen analysis.

In the preliminary model I will seek to capture only the first two of these observations. This will mean that the resulting model will be far from complete, also in the formal sense of not being able to analyze every grammatical sentence. A human without the facility for reanalysis would simply have no way of analyzing garden path sentences. Still, such a model is of some interest. The model may be more suitable in modeling the language processing in young children. Although not a main finding, research by (Trueswell, Sekerina, Hill, & Logrip, 1999) has indicated that young children are have a limited ability to recover from garden paths. The study in question is elaborated in section 7.3. Additionally, the preliminary architechture serves as a starting point for extensions in the following chapters.

4.2.1 Choice of functor

Choosing a functor will specify important structural characteristics of the process. First, there should be input, discretized on the level of words. Let W be an extensional dictionary of some language, i.e. every inflection of every word in some language. The functor will involve an exponent, and be on the form given below, where F is to be decided.

$$X \mapsto (F(X))^W$$

Second, there should be a way of returning an error when a word cannot be made to fit with the chosen analysis. Errors are in a set E. For the time being, there will be only one error, $fail$, thus $E := \{fail\}$.

$$X \mapsto (E + F'(X))^W$$

Moreover, with incremental left-to-right syntactic parsing there are only so many ways of integrating an input word with an existing syntactic structure. There should be a finite set of syntactic labels Syn characterizing how a given input was incorporated syntactically. The most basic case could be to have the labels be part-of-speech tags assigned to a word as well as a boolean indicating whether or not it was incorporated into the existing structure.

Additionally, encoding the seriality finding, there should be no occurence of products of the form $X \times X$ nor of powerset-functors ($\wp(X)$).

Define a functor $G : Set \to Set$ by the following assignment:

$$X \mapsto (E + (Syn \times X))^W$$

An G-coalgebra with carrier S is a (total) function:

$$co : S \to (E + (Syn \times S))^W$$

. A G-coalgebra co will then associate a state $s \in S$ with a function $co(s) : W \to E + Syn \times P$ such that for each word $w \in W$, either:

- $co(s)(w) \in Syn \times S$ in which case a label, together with a new state is determined in response to the word. This case is denoted an *increment*.

- $co(s)(w) = syn\_err \in E$ in case the mechanism fails to associate new states; there is an error relating to syntax.

Increments represent ordinary operation, where a change to the mental state can be made in response to the input word. The stop-case is reserved for the conditions where no further progress can be made, and extra processing is required for the mechanism to recover. Such a construction is similar to exceptions in object oriented programming, and is represented in much the same way by Jacobs (1995).

It is important that it is the sum $(E+(Syn \times X))$ that is exponentiated, and not just $Syn \times X$. A functor

$$X \mapsto E + ((Syn \times X)^W)$$

Would mean that a given state is either a syn_err or a state where any word determines a label and a new state. This situation would be unfortunate as it would be necessary to include dummy / error states mixed in with the proper states to represent an error that results from inputing only one of the many possible words.

4.2.2 Choice of carrier

The states of the mechanism should correspond roughly to the aspects of mental states of human that determine incremental syntactic decisions. In the preliminary model, I will let the carrier be a set representing syntactic information. This contrasts with constraint based models of parsing. These models claim that syntactic decisions are made using many different types of information. It is however also in contrast to two stage theories of parsing. Information about the analysis given to syntactic properties of preceding input should be sufficient to decide on an updated syntactic structure according to such models. However, other types of information, such as thematic roles, is thought to influence the propensity for an error with subsequent reanalysis[1]. In the following chapters, I will update the model with additional information included in states of the carrier.

Let P a set of all developing and complete incremental parses (spanning no more than a sentence). No assumption is made with respect to the grammatical formalism to be employed, as such decisions are outside the scope of the present project.

The preliminary model will be a coalgebra $incr$ on the functor G with carrier P.

$$incr : P \to (E + (Syn \times P))^W$$

There should be a monotonicity-requirement on $incr$. Let $\leq_P \subseteq P \times P$ be a substructure relation on P. The monotonicity requirement on $incr$ is that for all $p \in P$, for all $w \in W$

$$incr(p)(w) = (l, p') \in (Syn \times P) \Rightarrow p \leq_P p'$$

The precise meaning of $\leq_P$ depends on the carrier S, but the general idea is enforce the require-

---

[1] See for instance the sausage machine model of Frazier and J. D. Fodor (1978)

ment that states truly must be incremented. The carrier will be discussed further in the next section.

A distinguished empty syntactic structure $empty$ is however supposed to exist. Processing a sentence should now start by $incr(empty)(w_1)$ where w_1 is the first word of a sentence. Additionally for all p such that p is a complete grammatical analysis of a sentence $incr(p)(".")$ should be associated with a pair $(ok, empty)$.

A special type of deterministic incremental parser as defined by Nivre (2008) can be represented as a G'-coalgebra, where G' only differs from G in the set of labels. This formalism is introduced here to provide comparison and contrast with the present model, showing that we might consider the mechanism as a special case of an existing formalism. Moreover, some aspects of the Nivre (2008) formalism will be discussed when reprocessing is considered.

4.2.3 Incremental parsing formalism

Nivre (2008) builds a deterministic, incremental parsing algorithm on an underlying transition system. I will show how a specific kind of transition system can be converted into a coalgebra on a functor very similar to G.

Definition 4.2.1 (Transition system)**.** Nivre (2008) defines a *transition system* as a formalism for representing various algorithms that build dependency graphs in an incremental way. The definition from p. 518 in Nivre (2008) is presented in a slightly adjusted way here. A transition system is a 4-tuple $S = (C, T, i, C_t)$

1. C set of parser configurations. These are required to contain a list of the remaining words of the sentence, together with a syntactic representation. The syntactic component will be denoted by the name par, while the buffer of the remaining words will be denoted by buf. In Nivre (2008), syntactic representations are required to be sets of dependency arcs, forming the rudiments of dependency graphs.

2. $T \subseteq C^C$ of partial transition functions.

3. If L is a set of sentences, $i : L \to C$ is a function assigning to each sentence an initial parser configuration with an empty syntactic representation and the sentence as buffer. As such, $i(L) \subseteq C$ are the initial states of the parser, indexed by sentences.

4. $C_t \subseteq C$ set of terminal configurations.

In transition-based parsing, parser states are denoted parser configurations (Nivre, 2008, p. 518). However, such a transition system differs from the coalgebra being developed here in important ways. First, there are no constraints on when and how the buffer is emptied, meaning transitions do not reflect incremental processing on a word by word basis from left to right. A second way in which processing diverges from the present model is that there is no requirement that transitions be monotonic with respect to syntactic representations. I will define transition systems with these properties below:

Definition 4.2.2 (Left-right incremental transition system)**.** I define a left-right incremental transition system $S = (C, T, i, C_t)$ as one which is:
Left-to-right:
The buffer should be emptied from left to right.

$$\forall t \in T, (par, buf) \in C(\text{ t defined } \rightarrow t(par, buf) = (par', rest(buf)))$$

Monotonicity:
The monotonicity of the syntactic representation with respect to a substructure relation on such structures is important in order to ensure incrementality in the sense that syntactic decisions are final once made (p. 519). Let sub be such a substructure relation.

$$\forall t \in T, (par, buf) \in C(\text{ t defined } \rightarrow ((t(par, buf) = t(par', buf)) \rightarrow parsubpar'))$$

Such a definition is alluded to, but not made explicit in Nivre (2008, p. 519).

A way in which left-right incremental transition systems differ from the present model is that there is a requirement that parser states contain a list of remaining terminal nodes, in our case, words. This fact means the parser configurations / parser states differ depending on as of yet unprocessed input, meaning syntactic transitions do not exclusively depend on the syntactic structure thus far and the next word. In the present model, parser states are thought to model states of the language comprehender, and such states obviously do not contain the words that are not yet presented. There is little evidence that words are buffered before they receive syntactic processing. I will impose a rest-obliviousness condition on transition systems in order to avoid this complication.

Definition 4.2.3 (Rest-oblivious transition system)**.** A left-to-right incremental transition system $S = (C, T, i, C_t)$ is rest-oblivious if for all pairs of configurations $c = (par, buf), c' = (par', buf') \in C$, if $par = par' \wedge head(buf) = head(buf')$ then $\forall t \in T(t(c)defined \leftrightarrow t(c')defined \wedge (t(c), t(c')defined \rightarrow t(c') = t(c)))$

First, I define a functor where a rest-oblivious transition system can be represented as a coalgebra. A rest-oblivious transition system, may be represented by a coalgebra of the functor H given by:

$$X \mapsto E + \wp(T \times X)^W$$

In order to define the carrier of the H coalgebra on which the rest-oblivious transition system will be defined, I will introduce a relation $R \subseteq C \times C$. The relation will be used to group states with identical syntactic representations.

R is defined for all $(par, buf), (par', buf') \in C$:

$$R((par, buf), (par', buf')) \Leftrightarrow par = par'$$

A left-right incremental, rest-oblivious transition system S may be represented by a H-coalgebra $(\wp(C/R), p)$ in the following way. As carrier, I use the quotient construction C/R, defined by:

$$C/R := \{X \subseteq C : \exists x \in C(E_x^R = X)\}$$

Note that under this quotient construction, we have that $i(S) \in C/R$ as initial states as defined by Nivre (2008) only differ in their buffers.

The introduction of E comes as a consequence of the broadened set of inputs. Nivre (2008) provides as buffers only sequences of inputs that form grammatical sentences. Moving to the more general class of all kinds of sequences of words it becomes necessary to introduce an error-case, as new dead ends are introduced. That is, for some syntactic state par, there may exist a word $w \in W$ such that for no configuration $(par, buf) \in C$ is it the case that $head(buf) = w$. In that case, it is not grammatically possible for w to be included in the representation, and an error $ungram \in E$ will be thrown.

$$poss : C/R \to (E + \wp(T \times C/R))^W$$

Given for all $C' \in C/R$:

$$poss(C')(w) := \begin{cases} ungram \in E & \text{if } \neg\exists c = (par, buf) \in C'(head(buf) = w) \\ undef \in E & \text{if } \exists c = (par, buf) \in C'(head(buf) = w) \wedge tCs = \varnothing \\ tCs & \text{otherwise} \\ \quad \text{where } tCs := \{(t, C'') \in T \times C : t \in T \ \wedge t(c) \text{ defined } \wedge C'' = E_{t(c)}^R\} \end{cases}$$

By the axiom of choice one can select $c = (par, buf)$ such that $head(buf) = w$. As syntactic components for all configurations $c \in C'$ are identical, and the next element in the buffer is identical for all $c = (par, buf)$ where $head(buf) = w$, the syntactic component of $t(c)$ will be identical for all such c. Thus $E_{t(c)}^R$ is uniquely defined.

In case of an error $undef$, some configuration will have w as the next element of the buffer, but no transition functions are defined on this state.

The coalgebra does not explicitly represent terminal sequences C_t for reasons of brevity. We might define the terminal states of C/R as those C' where $\exists x(x \in C' \wedge x \in C_t)$

For any $poss(C') \in \wp T \times C'$ there is an associated injection $s : poss(C') \to T$ given by the assignment $s(t, C'') := t$ for all $(t, C'') \in poss(C')$. In essence, this means that labels uniquely determine transitions.

This transition system is nondeterministic, meaning there are a multitude of ways in which a given parser configuration may be extended to cover an additional input word (each labelled by a unique T). In Nivre (2008), the transition system is made deterministic by an *oracle*, mapping configurations to transition-labels.

$$o : C \to T$$

A deterministic, incremental parser is obtained by iterative application of the transition suggested by the oracle to a configuration c, starting for each sentence $l \in L$ in $i(l)$, and stopping whenever the configuration is terminal ($c' \in C_t$) (p. 519). Possibly however, o maps some $c \in C, c \notin C_t$ to $t \in T$ such that $t(c)$ is not defined, perhaps due to the possibility that there exists no $t \in T$ defined on c, in which case the incremental deterministic parser will halt.

It is worth noting that such an oracle in principle can peek at input from the remainder of the sentence in order to determine the transitions. As synactic processing occurs very quickly after words are presented, a rest-oblivious condition must also be imposed on oracles.

An oracle for the coalgebraic underlying transition system may be defined as $o' : C/R \to T$ in much the same way as for transition systems. However, any oracle for a transition system will not do. The rest-obliviousness-condition on the transition system ensured that exactly the same transitions are possible when two states only differ on the rest of the buffer. Still however, the oracle may choose a different transition depending on the rest of the buffer. A rest-oblivious oracle may simply be defined as an oracle $o : C \to T$ subject to:

$$\forall c = (par, buf), c' = (par', buf') \in C((par = par \wedge head(buf) = head(buf')) \to o(c) = o(c'))$$

In which case I define o' by o with the assignment $o'(C') := o(c)$ where $c \in C'$ is obtained by the axiom of choice.

The target coalgebra will be defined on a new functor G', which has as labels T instead of Syn, but is in other respects identical to G. G' defined below.

$$X \mapsto (E + (T \times X))^W$$

The H-coalgebra $(C/R, poss)$ together with an oracle now yields a G'-coalgebra $dinc$ with carrier C/R.

$$dinc(C') := \begin{cases} error \in E & \text{if } p(C') = error \\ o\_error \in E & \text{if } o(t) \notin s(poss(C')) \\ C'' \text{ where } (o(C'), C'') & \\ \quad \text{is given by inj. of } s. & \text{otherwise} \end{cases}$$

Note that there is a special case o_error for the case where the original oracle o assigns $c \in C$ to $t \in T$ where $t(c)$ is undefined. In the coalgebra above, the problem is explicitly handled by termination with a specific error-message.

4.2.4 Simplified procedure for processing sentences

I propose the following simplified iterative algorithm when processing sentences. Although not entirely psychologically plausible, it serves as a good starting point for further discussion of such procedures.

Assume a G-coalgebra $(P, incr)$. Let $[w_1, w_2, ...w_n]$ be a list of words, possibly consisting of several sentences. For practical reasons, the end of a sentence is signified by the period symbol. In its most basic incarnation, processing this list of a words is a matter of recursive application of $incr$, beginning at some start state. Define functor $I : Set \to Set$ given by the assignment:

$$I \mapsto ((Syn^* \times E) + (Syn^* \times X))^{W^*}$$

Where Syn^*, W^* are the sets of all lists over Syn and W respectively. From the coalgebra $(P, incr)$ we may now construct a H-coalgebra $(P, incr\_iter)$. The name $incr\_iter$ is chosen as its behavior is given by iteratively applying the underlying coalgebra $incr$.

Define recursive function $incr\_iter : P \to ((Syn^* \times E) + (Syn^* \times P))^{W^*}$

$$incr\_iter(p) : W* \to (E + (Syn^* \times P)) \text{ given by:} \tag{4.1}$$

$$incr\_iter(p)([]_W) \coloneqq ([]_{Syn}, p) \tag{4.2}$$

$$incr\_iter(p)(ws) \coloneqq \begin{cases} ([]_{Syn}, syn\_err) & \text{if } incr(head(ws)) = err \in E \\ (conc(syns, [syn]), x) \text{ where } incr(c,p)(head(ws)) = (syn, p') \wedge & \\ \quad incr\_iter(p')(rest(ws)) = (syns, x) & \text{otherwise} \end{cases} \tag{4.3}$$

$incr\_iter$ simply applies $incr$ recursively to a list of words, starting at some state p, and for each new word, either updating the state p or stopping if $incr$ stops. Labels (syntactic commitments) are recorded in a list, starting with the first-encountered label, ending with the last encountered label. Note that in the recursive case, we are not guaranteed either $x \in E$ or $x \in P$, as it is not known prior to processing the rest of the list. In contrast to $incr$, $incr\_iter$ provides a list of labels in addition to the error when there is no next state. In a sense, we get an extended error report compared to $incr$. Omitting the list of labels from the extended error would mean that case distinctions have to be made on the basis of where $incr\_iter$ ends up after having processed the rest of the list of words in order to determine output.

$incr$ is a mechanism that picks the preferred incremental analysis, and fails if whenever such an analysis is syntactically impossible. $incr\_iter$ is a mechanism that attempts to incrementally analyze a sequence of words, possibly spanning over more than one sentence, taking at each increment the preferred analysis, failing whenever the basic incremental mechanism $incr$ fails. One can think of $incr\_iter$ as the simplest possible control structure on a more basic process $incr$. As such, it has no way of recovering from garden paths, and simply gives up. More complex control structures that attempt to recover from error by reanalysis will be discussed in section 4.3.

The present control structure has some plausibility in young children, shown by experiment to comprehend language without much reprocessing, fixating on initially chosen analyses Trueswell et al. (1999), see section 7.3. There, five year olds were unable to revise analyses, and instead stuck to their initial interpretations. Transitioning from a very simple control structure to a more complex control structure that permits syntactic reanalysis may be a way in which children extend grammatical coverage, but this possibility is not explored further here.

4.3 Reanalysis

The preliminary mechanism proposed in section 4.2.4 has obvious problems with completeness. When initial commitments are locally acceptable, but globally unacceptable, the mechanism returns an error. Modern psycholinguistic theories of parsing agree that some sort of extra processing stage, often denoted reprocessing or reanalysis should kick in when an analysis becomes untenable. Two-stage theories like the garden path model and constraint based models differ somewhat in their predictions on when reprocessing occurs. The different views, and evidence ruling in favor of the garden path prediction are presented in section 4.3.1.

When it comes to reprocessing proper, and not just surrounding conditions, there is less evidence (Pickering & R. P. G. van Gompel, 2006, p. 478). That is, what syntactic commitments should the processor renege on? There is some evidence ruling out the most simplistic of algorithms, such as reevaluating decisions backwards from the point where an error was discovered, and starting at the beginning and choosing differently from the very start (Frazier & Rayner, 1982). Reanalysis likely imposes demands on working memory, and comprehenders with different working memory capacities might employ different algorithms. An added complication is the fact that with auditorily presented language, must be stored in memory in orded to be reanalyzed. In reading, where the majority of reprocessing research is done, comprehenders need not consult their memories of the preceding linguistic input, but can read it anew. Thus, reprocessing findings may not apply equally to auditorily presented language.

Further complications in reprocessing comes from the fact that several studies have found evidence that discarded interpretations still influence processing (Christianson, Hollingworth, Halliwell, & Ferreira, 2001; Sturt, 2007; R. P. van Gompel, Pickering, Pearson, & Jacob, 2006, e.g.).

Patson, Darowski, Moon, and Ferreira (2009) replicated the findings of Christianson et al. (2001), but with a stronger paradigm. Patson et al. asked readers were asked to paraphrase the meaning of garden path sentences (e.g. (3-a)), or variants with a disambiguating comma (3-b).

(3) a. While Anna bathed the baby spit up on the bed.
b. While Anna bathed, the baby spit up on the bed.

Participants were much more likely to produce a paraphrase that had elements of the discarded analysis, e.g. “Anna bathed the baby and it spit up on the bed.” in the garden path condition than in the comma condition. Reanalysis appears not to be a clean, straightforward procedure. The algorithm for reanalysis presented in section 4.3.2 departs from psycholinguistic evidence, and represents naïve and idealized reprocessing, although inspired by psycholinguistic evidence.

4.3.1 Conditions determining reanalysis

A sequence of words will sometimes end up in an error state when processed by repeated application of $incr$ (i.e. $incr\_iter$). At this point, reanalysis is needed. However, there may be other conditions under which reanalysis should happen. Constraint based and garden path models of parsing make differing predictions as to the conditions when reanalysis takes place.

The structure of processing proposed by garden path theories constrain reanalysis to cases where initially chosen analyses prove troublesome when further words lead to syntactic or semantic problems (although semantic processing is not discussed until later). A stronger claim is the Reanalysis As a Last Resort hypothesis, where comprehenders are thought not to reanalyze sentences unless there is no possible continuation (J. D. Fodor & Frazier, 1980, p. 427).

The conditions under which garden path models predict reanalysis are contained in the set of instances under which constraint based models predict reanalysis. Additionally however, constraint based models predict that factors external to the syntactic and semantic acceptability of a chosen analysis can cause reanalysis. Among such factors is how biased a verb is towards the chosen reading.

Sturt, Pickering, Scheepers, and Crocker (2001) investigated the effect of manipulations of the bias of a verb towards the chosen analysis on the propensity of participants to reanalyze sentences. Sturt et al. (2001) conducted a corpus study, and found pairs of verbs such as “found” and “discovered” that were different in their preference for direct object vs. other readings of immediately subsequent NPs. For instance, “found” is strongly biased towards the direct object reading, wheras “discovered” is weakly biased towards this reading. From such pairs of words, pairs of sentences such as the following were constructed.

(4) a. The troops who discovered the enemy spy had used up all the supplies were later mentioned in the press report.
  b. The troops who discovered *that* the enemy spy had used up all the supplies were later mentioned in the press report.

(5) a. The troops who found the enemy spy had used up all the supplies were later mentioned in the press report.
  b. The troops who found *that* the enemy spy had used up all the supplies were later mentioned in the press report.

Sentences (4-a) and (5-a) are locally ambiguous because “The troops who discovered the enemy spy" in (4), and "The troops who found the enemy spy" have readings of the enemy spy as the object of "discovered" or "found" respectively, and as the beginning of a complement. There are, according to the authors, good reasons to believe that the direct object reading is the initially chosen for (4-a) and (5-a). For sentences (4-b) and (5-b) however, a complement reading is forced by the complementizer "that".

A prediction of the constraint based view is that verb bias of the chosen analysis should influence whether or not comprehenders reanalyze a sentence. If this is the case, "had used up" should take longer to read in the ambiguous condition than in the unambiguous condition when the verb bias is weak ((4-a) vs. (4-b)), compared to when the verb bias in favor of the chosen analysis is strong ((5-a) vs. (5-b)) (p. 294). In the unambiguous condition, no possibility for reanalysis exists at "had used up". Sturt et al. (2001) did not find an interaction of verb bias with ambiguity (p. 295). Similar findings were obtained by D. Schneider and Phillips (2001). Still, compared to other aspects of syntactic processing, the conditions determining reprocessing have not been thoroughly explored.

Following these findings, it appears approriate to code the situations that need reanalysis as sequences of words that end up in E. The simple dynamics outlined in section do nothing when we end up in E. A more complex dynamics for processing sequences of words is required.

4.3.2 Reanalysis in the model

As it stands at present, error reporting is coded on the coalgebra $incr$. That is, in a given state p, for a given word w, the mechanism may either determine a new state or it may determine a new state at which point an error is reported. The main idea of the present reanalysis algorithm is to handle errors by making a different decision at some previous choice point, and to restart the basic mechanism from there. In order to accomplish this behavior, the incremental mechanism $incr$ must be situated in a transition space.

Recall that in the discussion of the incremental parsing algorithms defined by Nivre (2008) (section 4.2.3), an underlying transition space was used. In the present case, a richer underlying transition space will be a coalgebra on a functor J:

$$X \mapsto E + (\wp((0,1) \times Syn \times X))^W$$

An additional output from the set of all probabilities, the closed interval $(0,1)$ is added in order to rank alternatives. A simple list is not used, as comparability across sets of alternatives is desired. We consider J-coalgebras (A, f) where for all $a \in A$, for all $w \in W$ it is the case that if $f(a)(w) \in \wp((0,1) \times Syn \times A)$ then $r_2(f(a)(w))$ is injective. r_2 is the projection of the second coordinate from $f(a)(w) \subseteq (0,1) \times Syn \times A$. That is, for any label syn there is maximally one $(r, syn, a) \in f(a)(w)$. Labels, if they apply, uniquely identify a successor state. An underlying transition space with ranked alternatives is now a HR-coalgebra $(P, incr\_alt)$. Assume that for no (p, w) is $incr\_alt(p)(w) = \varnothing$. In cases where there are no alternatives, $incr\_alt(p)(w) = \varnothing$

should end up in E. Moreover, we want values of $(0,1)$ to be probabilities for the respective alternatives, requiring that they sum to 1 $p \in P, w \in W$:

$$\Sigma(r_1(\mathit{incr\_alt}(p)(w))) = 1$$

Where $r_1(r, syn, p) = prob$ for all $(r, syn, p) \in (0,1) \times Syn \times P$. Note that this requirement also excludes the possibility of an empty set of further transitions, as such a set would sum to 0. I do not make assumptions regarding the meaning of the probabilities used here. Their main use is to rank alternatives in a way which is comparable between different syntactic decision-points. Again, the carrier is P. We can obtain a coalgebra similar to $incr$, denoted $\mathit{incr\_max}$ on the functor K. $\mathit{incr\_max}$ differs from $incr$ in that in addition to the syntactic label of the transitions, in that the set of non-chosen alternatives $Alt \in \wp((0,1) \times Syn \times P)$ will also be part of the output. The functor K is defined below:

$$X \mapsto (E + (\wp((0,1) \times Syn \times X) \times Syn \times X))^W$$

Define K-coalgebra $(P, \mathit{incr\_max})$ given for all $p \in P$ by:

$$\mathit{incr\_max}(p)(w) := \begin{cases} \mathit{syn\_err} \in E & \text{if } \mathit{incr\_alt}(p)(w) = err \in E \\ (Alt, (syn, p')) \text{ where } \exists r\big((r, syn, p') \in \mathit{incr\_alt}(p)(w) \wedge & \\ \quad \forall (x, y, z) \in \mathit{incr\_alt}(p)(w) : r > x \quad \wedge & \\ \quad Alt = \mathit{incr\_alt}(p)(w) \smallsetminus \{(r, syn, p')\}\big) & \text{otherwise} \end{cases}$$

Essentially, $\mathit{incr\_max}$ is identical to $incr$, save from the fact that it gives off an output of ranked alternative continuations. There is some evidence that at least some alternative continuations are activated in incremental processing (Cai, Sturt, & Pickering, 2012), but this phenomenon has not been explored in detail.

Note that in $\mathit{incr\_max}$ it is asssumed that maximality uniquely identifies (r, syn, p'), which need not mathematically be the case. At this point, we are ready to define more advanced behavior in case there is an error. The situation in $\mathit{incr\_alt}$ is that we are presented with ranked alternative continuations of sentences, indexed by labels. The situation presents as an informed search problem. However, human language comprehension appears to differ from the standard search algorithms. In algorithms such as A^*, the search algorithm decides what node to expand next. The present mechanism has a two-part design, a relatively simplistic mechanism that deterministically chooses a path through the search space, and a recovery mechanism that starts the simplistic mechanism off on a new path if the chosen path does not succeed. The simplistic mechanism is represented by $(P, \mathit{incr\_max})$, while the recovery mechanism is represented below.

Recall the functor I, as given in 4.2.4. On it, the coalgebra $\mathit{incr\_iter}$ was defined on lists of words by incremental application of $incr$, stopping whenever $incr$ stops. It will be assumed that $incr$ can be derived from $incr\_max$. A control structure on $incr\_max$ that reanalyzes when it fails will have the following behavior (omitting some technical details). Starting from a list of words.

- Do $incr\_max$ on each word, storing alternatives and the list of labels as each are generated by $incr\_max$.
- If the list of words is empty, then we return the list of labels of the transitions that got the algorithm there, the final state is the final state of $incr\_max$.
- If $incr\_max$ returns an error upon trying to increment with the next word, do reanalysis. Choose the most promising of the stored alternatives, and repeat the procedure from that point, keeping the list of remaining alternatives.

More formally, the procedure can be encoded as a coalgebra. Let L be an endofunctor on set, given by:

$$X \mapsto (E + (Syn^* \times X))^{(W^*)}$$

Before I define the reanalysis algorithm as an L-coalgebra, I introduce the components of the carrier of this coalgebra. A list of all the words to be processed will be part of the carrier, hence W^* is in the carrier. The purpose of this list will be to be able to backtrack from a given position in the list. Second, an element of $\mathbb{N}$ keeps track of the position in the list, so when alternatives are created, they can be tagged with the position from which reanalysis should be done. A function:

$$listfrom : W^* \times \mathbb{N} \to W^*$$

Gives for each n, list of words ws the list of words ws' after removing the first n elements from the list. If n exceeds the number of elements in ws, return the empty list $[]$. A set

$$\wp((0,1) \times P \times \mathbb{N} \times Syn^*)$$

Stores the alternatives of the reprocessing algorithm. Alternatives are stored as parser states with probabilities, the number of words that have processed from the list of all words to get to this point, and the list of labels leading to this alternative.

Define L-coalgebra:

$$\begin{aligned} incr\_rean :& W^* \times \mathbb{N} \times \wp((0,1) \times P \times \mathbb{N} \times Syn^*) \times Syn^* \times P \\ &\to (E + (L^* \times (W^* \times \mathbb{N} \times \wp((0,1) \times P \times \mathbb{N} \times Syn^*) \times Syn^* \times P))^{W^*} \end{aligned}$$

$$incr\_rean(all, n, AltNL, syns, p)(nil) := (syns, all, n, AltNL, syns, p)$$

$$incr\_rean(all, n, AltNL, syns, p)(ws) :=$$

$$\begin{cases} incr\_rean(all, n+1, AltNL', conc(syns, [syn]), p')(rest(ws)) \\ \quad \text{where } AltNL' := \begin{cases} AltNL + (Alt \times \{n\} \times syns) & \text{if } head(ws) \neq \text{"."} \\ (Alt \times \{n\} \times syns) & \text{otherwise} \end{cases} \\ \quad \text{if } incr\_max(p)(head(ws)) = (Alt, syn, p') \\ \\ incr\_rean(all, m, AltNL', conc(syns, [syn]), p')(listfrom(all, m)) \\ \quad \text{where } choose(AltNL) = (AltNL', ((syns, p'), m, syns)) \\ \quad \text{if } incr\_max(p)(head(ws)) = stop \in E \wedge AltNL \neq \varnothing \\ \\ syn\_err \in E \\ \quad \text{if } incr\_max(p)(head(ws)) = stop \in E \wedge AltNL = \varnothing \end{cases}$$

The algorithm is described informally below. The basic case is that there are no more words, evidently $incr\_max$ was able to find a way of incrementally analyze the sentence. The list of syntactic labels associated with transitions leading there is returned, and the present state is the state we end up in. Note that if the list of word contains multiple sentences, only the syntactic analysis pertaining to the very last one will be stored in p. An accumulator for such states can be added, but was not for reasons of brevity. The list of labels is at present the only way of recovering this information.

A list all of all of the words in the list of words is kept constant for backtracking purposes, ensuring that the mechanism can backtrack from any point in this list at a later time. n keeps track of the current position in the list. $AltNL$ is a set of alternatives, where each alternative is tagged with both the position in the list where one would need to continue from, and the list of labels leading to the alternative. $labels$ is the current list of labels, and may be completely overwritten if the sentence is reanalyzed.

When the list of words is not empty, there are three cases. First, if $incr\_max$ does not return an error, and the word was not ".", simply go ahead and add generated alternatives to the set of alternatives, making sure to tag them with what position in the list we are at. In the future, if reanalysis is needed, we will be able to start from the right word. Moreover, alternatives need to be tagged with the list of labels leading them. If reanalysis is needed, this list of labels will have to be continued in order to build the output. This ensures that the labels that are part of the output will be those that are actually part of the chosen path, and not of disregarded analyses. If the word on the top of the list of words to be processed was indeed ".", we will not consider alternative readings of this sentence, as it has been successfully analyzed. Hence, the existing set of alternatives must be forgotten.

Second, if $incr\_max$ does return an error, it is necessary to do reanalysis. The auxillary function $choose$ takes as input a set of alternatives tagged with position and the list leading there, and returns 1. an updated set of alternatives, 2. a chosen alternative, where reanalysis will start from. Then, the procedure is repeated from this alternative state. I assume that $choose$ picks from the list of alternatives the most probable one.

4.3.3 Evaluation

There are cases where the model of reanalysis will not initially decide on the most frequent reading of a sentence. It behaves in a greedy way, maximizing the probability of initial syntactic commitments. However, such an initial commitment may force the processor into a situation where it has to adopt less common syntactic choices. Essentially, the global optimum in terms of corpus level probability may well be missed.

However, this situation is not all negative. If the present model of reanalysis indeed has a practically significant degree of correspondence to the order in which humans consider readings of sentences, it should have applications in evaluating the readability of text.

Chapter 5

Extending the model with representations of meaning

In the present chapter, I argue that the carrier (state space) of the preliminary model should be extended with (incrementally built) representations of meaning, in addition to the existing parser states (5.1). I define an extended model in section 5.2.

A major discussion in psycholinguistics, the debate on whether incrementally constructed meaning representations influence further syntactic decisions, can now be represented by the model. I argue that there is not sufficient evidence to believe that the incrementally built meaning representation influence further syntactic decisions (5.4). Rather, meaning representations resulting from choosing an analysis appears to cause errors, leading to reprocessing.

As such, the model extended with meaning representations is formally speaking a refinement of the previous model. The value of the extension for NLP is primarily to discard inappropriate readings at an early stage, and prompt reprocessing. The relationship between the preliminary model and the model with meaning representations is represented formally in section 5.5. Finally, an extended reanalysis algorithm is defined in section 5.6.

5.1 Psycholinguistic evidence on the carrier

In the present section, I introduce an elementary finding on the time course of language comprehension, the N400 effect. I argue that elementary findings on the N400 effect establishes the involvement of meaning representations during incremental language comprehension. In terms of the preliminary model, such evidence allows us to specify an extended carrier.

5.1.1 The N400

In research using *Event Related Potentials* (ERP), the electrical field on the scalp following an event is averaged over many trials and participants (Ward, 2010, p. 38). This research technique has a high temporal resolution, meaning that it can pinpoint *when* the brain events corresponding to patterns of electrical activity happened. The N400 response is a negative signal with a peak at about 400 ms after the presentation of a stimuli (Kutas & Federmeier, 2000, p. 464). In the present text, N400 responses to words (possibly with different modalities of presentation) will be discussed, although effects have been observed for other meaningful stimuli (Kutas & Federmeier, 2009, p. 628).

When stimuli belonging to two different experimental conditions elicit differing N400 responses, we say that there is an N400 effect associated with the experimental conditions. Note that the onset of the N400 response comes long before 400 ms, according to Kutas and van Petten (1994) “In most experiments, N400 amplitude differences are apparent by 200 ms post-stimulus.” (p. 106). The N400 response is responsive to manipulations of the congruity of a word with its surrounding context.

In early research, Kutas and Hillyard (1980) found a greater N400 response to semantically incongruent words embedded in sentences than to semantically expected / congruent words. In sentences such as “He took a sip from the transmitter”, the incongruent word (here “transmitter”) elicited a stronger N400 than was elicited by expected / congruent words such as “sugar” in the sentence “I take coffee with cream and sugar”. Congruent but unexpected words, e.g.”waterfall” in “He took a sip from the waterfall” also elicited greater N400 amplitudes than congruent but expected words, but far from as great as the incongruent words.

van Petten and Kutas (1991) presented regular sentences, nonsensical grammatical sentences and randomly assembled word sequences to participants, and found that for open-class words, the associated N400 amplitude was lower for words in increasing positions in a sentence only for the regular sentences. With regular sentences only, an increasing sentence position was thought to be associated with an increasing semantic context. The result indicate that the N400 is not only sensitive to semantic associations of words, but to a developing meaning representation. No differential effect of sentence position was found on N400 amplitudes for open-class words in the grammatical vs. random conditions, indicating that the result was not due to syntactic context.

Kutas and Federmeier (2009) are careful to point out that

> ... although ERP parameters are sensitive to psychological variables, they are neither generally nor readily reducible to psychological constructs. Ultimately, it is the brain’s “view” of cognitive processing that we seek to characterize. (p. 624)

For instance, it is not in general the case that the the N400 is responsive to the truth of a sentence. Fischler, Bloom, Childers, Roucos, and Perry (1983) asked participants to indicate whether some sentences were true or not whilst monitoring their EEG. They found that for affirmative sentences on the form “A is B”, false sentences such as “A robin is a bird” elicited a greater N400 than true sentences such as “A robin is a tree”. For negated sentences on the form “A is not B” however, the N400 was greater for true sentences such as “A robin is not a tree” than for false sentences such as “A robin is not a bird”. Fischler et al. (1983, p. 406) explain the results as being due two a two step process where the sentence without the negation is understood in terms of semantic memory before its truth is evaluated. In the second step, the resulting truth value from the first process is reversed and represents the truth value of the negated sentence.

5.1.2 Coalgebraically

The experiments above may be understood in terms of coalgebra, following the example in Jacobs and Rutten (2011). Suppose we come by a mechanism where the inner states are hidden. The task is to characterize the nature of the inner states of the machine, at least the parts of the inner state of the machine that matter to how the machine behaves. We may enter into the machine input (a word), there is a reset button, and there are many measures that can be taken from the machine, such as how much time is used after a word is entered until a loud buzzing sound quiets down and the machine appears to be in a state where it is ready to accept a new input. We might consider such a measure a reading time (more abstractly a response time). After careful manipulation of the machine, we find that the loudness of a certain buzzing noise after an input differs according to what appears to be the semantic fit of the word with preceding words of the same sentence. The machine emits a louder particular buzzing noise the worse the fit is. Such a machine can be considered as a coalgebra on the following functor:

$$X \mapsto (Noise \times X)^W$$

$Noise$ is a set of positive reals, indexing loudness values for the particular kind of buzzing noises associated with a transition. We have a fixed but unknown state space S with a special state $r \in S$ which is obtained by pressing the restart-button. The behavior of the machine is encoded on a function $c : S \to (Noise \times S)^W$. Little is known about c.

We conclude that buzzing noises are indicative of the internal processes of the machine, and that when different buzzing durations occur, different results may well obtain inside the machine. The loudness of the buzzing noise appears to vary according to many other variables aswell, and so we must compare loudness for the same word varying input only in the crucial properties that pertain to the semantic fit of the critical word. Let $list_1, list_2$ be sequences of words such that the fit of a word w differs, with w fitting better in $list_1$ than in $list_2$. Let s_1, s_2 be the states resulting from first pressing the restart-button, and then iteratively entering the words in the respective lists. We observe that $r_1(c(s_1)(w)) < r_1(c(s_2)(w))$. Repeating the experiment for different pairs also manipulating semantic fit, results persist, leading us to further conclude that the inner state of the machine somehow records semantic properties of the preceding input belonging to the same sentence.

There exists two stronger, differing accounts of the N400 effect (Lau, Phillips, & Poeppel, 2008, p. 921), each with a substantial amount of supporting evidence. These accounts will be discussed in later chapters (6,7).

Conservatively, the mere existence N400 effect is evidence that at least a rudimentary meaning representation is incrementally generated, and that such a representation has an on-line influence on processing of subsequent from the same sentence. Let the set of all such representations be M. The new carrier then is $M \times P$. Correspondingly, the transition $incr$ must be changed in response to the altered inputs.

5.2 Extending the coalgebra with M

The present section is about the question of how to extend $incr$ once $M \times P$ becomes the carrier. Recall that M is the set of all developing and complete sentence level meaning representations.

In order to reflect the fact that comprehenders make incremental semantic commitments as they process language, the set of labels is now extended. Let Sem be a set of labels reflecting incremental semantic commitments. At minimum, I will require that Sem contains word senses. I will discuss evidence indicating that this is the case in the next chapter (6.4.1).

Recall that in the functor G on which $incr$ was defined in section 4.2.1, a set of labels Syn denoting syntactic commitments was fixed:

$$X \mapsto (E + (Syn \times X))^W$$

Therefore $incr2$ must be a coalgebra on a new functor $G2$, with a new set of labels.

$$X \mapsto (E + (Sem \times Syn \times M \times P))^W$$

Let $incr2$ be the new coalgebra:

$$incr2 : M \times P \to (E + (Sem \times Syn \times (M \times P)))^W$$

In the set of all developing and complete meaning representations, there should be a distinguisted representation $empty$. As with $incr$ and $p \in P$ when a sentence has finished processing:

$$incr2(m,p)(".") \in (Sem \times Syn \times (M \times P) \Rightarrow incr2(m,p)(".") = (ok, ok, empty, empty)$$

Similarly, the monotonicity requirement on P should be kept intact, and extended to apply to M as well. Let $\leq_M \subseteq M \times M$ and $\leq_P \subseteq P \times P$ be substructure-relations on M and P respectively. Then

$$\leq_{M\times P} := \{((m,p),(m',p') : m \leq_M m' \wedge p \leq_P p'\}$$

is a substructure-relation on $M \times P$. The corresponding monotonicity requirement on $incr2$ becomes:

$$incr2(m,p)(w) = (sem, syn, (m',p') \in Sem \times Syn \times (M \times P) \Rightarrow (m,p) \leq_{M\times P} (m',p')$$

In order to represent transitions by their labels, one might understand in this setting the labels as including information about semantic aspects of incremental processing, such as word senses assigned to incoming words.

A much debated issue in psycholinguistics can be understood as a question about $incr2$. To what extent are incremental syntactic decisions affected by meaning representations? That is, does the component and M affect the behavior of $incr2$ in terms of which next P-state is generated? If the component M of the carrier records meaning representations, it is possible that such information could cause incoming words to be syntactically incorporated in a way that makes the meaning representation plausible.

5.3 Relation to syntax and semantics in psycholinguistics

Interactive approaches to syntactic processing such as Marslen-Wilson (1975) and MacDonald et al. (1994) claim that many kinds of semantic factors can influence incremental syntactic decisions, including developing meaning representations. The pipelined, garden path approach of Frazier and Rayner (1982) claims that a incremental syntactic decisions are influenced by syntactic information exclusively. According to the garden path model, the component M has no say in deciding how the P component is incremented. That is, only P (and of course a word w) has anything to say about increments to P[1].

The garden path model views increments as consisting of two stages. First, the syntactic principles of minimal attachment and guide the selection of a increment to a syntactic structure. Note that the principles governing attachment of a new constituent are only sensitive to the part-of-speech of the constituent. In a second stage of processing, the semantic consequences (encoded on M) are computed. Reanalysis may occur as a consequence of either the first or of the second stage, if there is a syntactic or semantic problem respectively. Syntactic or semantic problems must be severe for reanalysis to occur.

Interative models make the opposite claim, that any of the components can influence incremental syntactic decisions. Although interactive models such as MacDonald et al. (1994) claim that increments consist of stages; activation of multiple representations and constraint based selection of a single alternative in each category, these stages are not in general split across syntactic and semantic processing.

---

[1] The garden path model makes an even stronger claim. Only the part-of-speech information belonging to words are thought to influence syntactic decisions. The claim appears to be that if two syntactic/parser components $p, p' \in P$ are identical apart from lexeme information (at least for open class words), and $w, w' \in W$ are closed class words or have identical part-of-speech tags, then the incremented syntactic structures will also be identical up to lexeme information for open class words. That is, decisions on how to increment a syntactic state should not be influenced by the identity of the next word, or indeed any variation in identity of the words in the P component. However, lexical information does indeed appear to influence ongoing incremental decisions, but the strength of this influence is in dispute (Pickering & R. P. G. van Gompel, 2006, p. 465).

5.3.1 A clarification on syntax and semantics in incremental parsing

In the study of human parsing, the question of the possible interaction of semantic and syntactic information has been hotly debated[2]. It appears that this question has two senses, which are not always carefully delineated.

1. What is the nature of the relationship between syntactic and semantic knowledge when language comprehension is considered as a mapping from word strings to meaning representations?
2. What is the nature of the relationship between syntactic and semantic knowledge in the intra-sentiental incremental steps of language comprehension?

The garden-path theory of Frazier and Rayner is a highly influential pipelined theory, wherein syntactic principles of minimal attachment and late closure govern incremental changes in syntactic states. The role of semantics is secondary, providing feedback that an incremental change to syntax may be erroneous (Eysenck and Keane, 2010, p. 378; R. van Gompel, 2006, p. 251). Note however that such a possibility entails that semantic information can interact with parsing in the first sense (sentence level), as backtracking may be prompted by semantic anomalies resulting from the chosen analysis. Indeed there are empirical findings corroborating this possibility (cf. Kuperberg, 2007). A resultant syntactic analysis of a sentence then, may in fact be decided upon using semantic knowledge in addition to syntactic knowledge. In the second sense, there is of course little interaction in garden path theories.

It appears that in human parsing, even the mainstream theoretical position closest to the standard pipeline, in fact contradicts it on the sentence level. The question being investigated in psycholinguistics is the question of the interaction of syntax and semantics in incremental stages of processing, assuming that such interactions do indeed happen on the sentence level.

5.4 Effect of meaning representations on incremental parsing

The question of the effect of meaning representations on incremental parsing is the question of how one (of many) sources of semantic information may influence further syntactic processing. According to two stage models such as the garden path model, syntactic processing happens is a seperate process that occurs before semantic processing and without semantic involvement.

The findings of Frazier and Rayner (1982) discussed in section 4.1.1 corroborate the garden path model, as the principles predicted that comprehenders would choose the inappropriate analysis for the garden path sentences, but not for non-garden path sentences. However, the evidence hardly rules out alternative explanations.

---

[2] See Pickering and R. P. G. van Gompel (2006) for an overview.

Stronger evidence obtained by Ferreira and Clifton (1986) indicates that the effects of selectional restrictions on incremental syntactic decisions is somewhat limited. Selectional restrictions denote the semantic properties that are required of the semantic arguments of verbs. Ferreira and Clifton (1986) tested the garden path model experimentally, by investigating whether or not the semantic content associated with different lexical items (encoded on M in the model) would influence the tendency of comprehenders to be led down the garden path.

(1) a. The defendant examined by the lawyer turned out to be unreliable.
b. The evidence examined by the lawyer turned out to be unreliable.

As part-of-speech sequences, both sentences admit local incremental analyses where "defendant" or "evidence" is the subject of "examined". Semantically speaking, "evidence" does not possess agency, and should rule out such a reading in (1-b). However, reading times as monitored by eyetracking around the syntactically disambiguating word "by" were the same in both cases, indicating that semantic information did not influence the incremental syntactic decision. Moreover, adding "that was" after "defendant" and "evidence" reduced reading times of "by", indicating that syntactic information could direct comprehenders syntactic decisions in a way that semantic information did not. In terms of the model, the question is whether the M component can be constructed before the P component, and if so if it can influence the P-component. In terms of the model, the question is whether the semantic properties of "The evidence" (encoded by $m \in M$), the absence of the animacy property, could prevent a subject reading with respect to "examined" in the subsequent P component. Typically, the subject of "examined" is agent, and the M component should have directed the parser towards a passive reading of "examined" if semantic properties of preceding input informed syntactic decisions. Similar findings, indicating that selectional restrictions do not guide syntactic decisions, have recently been made by Kizach, Nyvad, and Christensen (2013, Experiment 2).

More general findings speak to the order in which information is considered when incremental syntactic decisions are made. McElree and Griffith (1995, Experiment 1) measured reaction times for decisions about the well-formedness of sentences containing errors of syntactic and semantic types. Detections of grammatical errors pertaining to the syntactic category of a word (e.g. "Some people rarely books.") were made faster than subcategory violations (e.g. "Some people agree books.") which were made faster than errors pertaining to thematic roles ("Some people rarely alarm books."). These results seem to support the garden path model, as different types of linguistic information are considered in a sequence. Possibly, the syntactic information pertaining to a word will be available to influence incremental syntactic decisions on how it should be integrated into the developing structure before the semantic information pertaining to a word. The semantic information then, is simply not available to influence incremental syntactic decisions.

However, J. D. Fodor, Ni, Crain, and Shankweiler (1996) investigated information on the *availability* of information on the semantic fit and syntactic fit of a critical word with a preceding part of a sentence. Sample experimental materials from J. D. Fodor et al. (1996, p. 31) are presented below.

(2) a. It seems that the cats from across the road won't *eating* the food that Mary puts out on the porch every morning as soon as she gets up. (syntactic anomaly)
b. It seems that the cats from across the road won't *bake* the food that Mary puts out on the porch every morning as soon as she gets up. (semantic anomaly)
c. It seems that the cats from across the road won't *eat* the food that Mary puts out on the porch every morning as soon as she gets up. (control)

In experiment 2, participants were presented audio recordings of the experimental materials. A visually presented lexical decision task was presented slightly before, immediately after or slightly after the critical word was presented. The lexical decision task took longer for the syntactic and semantic anomalies[3] than for the control word only when it was given immediately after the critical word was presented. According to J. D. Fodor et al. (1996), the findings indicate that both incrementally produced syntactic and semantic information are immediately available to influence further processing and are not the cause of the delay in McElree and Griffith (1995). To explain the findings of McElree and Griffith (1995), J. D. Fodor et al. propose that the results indicating delayed detection of semantic errors relative to syntactic errors are due the architechture of the language processing system instead of differences in timing in the availability of different types of information. The results of McElree and Griffith (1995) were reproduced in Experiment 3. Participants took longer to indicate that an error had been made in the sentences containing semantic errors than in the sentences containing syntactic errors (pp. 44-45). Supporting the proposal of J. D. Fodor et al. (1996), there are findings that indicate that different brain areas are responsible syntactic and semantic processing (Friederici, 1995, 2012; Peelle, Cooke, Moore, Vesely, & Grossman, 2007).

Friederici, Pfeifer, and Hahne (1993) investigated the time course of syntactic versus semantic processing using an ERP paradigm. Participants were presented auditorily with well formed sentences, sentences with phrase structure violations (i.e. syntactic category violations), sentences with morphological violations, or containing selectional constraint violations (semantic violations). The N400 effect was observed in response to the selectional constraint violations.

> The syntactic phrase structure errors realized as a violation of syntactic word category, in contrast, evoked an early negativity peaking around 180 ms with a maximum over frontal and anterior lateral electrode sites ...(p. 190)

Evidently, some syntactic processing relating to word category occured prior to the semantic processing associated with selectional constraints.

---

[3] J. D. Fodor et al. (1996) use the term *pragmatic* where I use semantic. I have changed the wording for a more coherent presentation.

Still, selectional restrictions are but one pathway whereby semantic information could theoretically influence incremental syntactic processing. Other theoretically possible pathways exist, and investigation of these pathways have yielded evidence of the influence of semantic information on incremental syntactic decisions, albeit in less direct ways. These are discussed in chapters 6 and 7.

In the next section, I discuss how these findings can be represented coalgebraically.

5.5 Representing the relation of $incr2$ to $incr$ formally

This section is about how one can represent the situation where meaning representations do not influence incremental syntactic decisions as a relation between $incr2$ and $incr$. I will assume that $incr2$ is subject to the following property with respect to $incr$. The idea is that in order to construct $incr2$ from $incr$, we do the following:

1. For every p,w where $incr(p)(w) \in E$, make sure that for all $m \in M$, $incr2(m,p)(w) = incr(p)(w) \in E$.

2. In case $incr(p)(w) \in Syn \times P$, then for a fixed $m \in M$ one of two things should happen.
   (a) $incr2(m,p)(w) = (sem, r_1(incr(p)(w)), m', r_2(incr(p)(w))) \in Sem \times Syn \times M \times P$
   (b) $incr2(m,p)(w) = sem\_err \in E$

Parser errors of $incr$ should be inherited by $incr2$, when there is no permitted syntactic increment, there should be no way of incrementing both the parser state and a meaning representation. If however, a transition of $incr$ does not lead to an error, $incr2$ could successfully make a transition, where the updated parser state and the syntactic label produced by $incr2$ are identical to those produced by $incr$. Additionally, $incr2$ has an updated meaning representation m', and semantic label sem. Alternatively, even if $incr$ does not lead to an error, $incr2$ might do so, on account of a failure to integrate a word into a meaning representation (e.g. violating selectional restrictions, implausibility).

Coalgebraically, we may encode this situation in the following way. Any $G2$-coalgebra can be translated into a G-coalgebra. Given a $G2$-coalgebra (A,c) associate a G-coalgebra $(A, forget\_Sem(c))$[4]. Defined for all $a \in A$:

$$forget\_Sem(c)(a)(w) := \begin{cases} c(a)(w) & \text{if } c(a)(w) \in E \\ (\pi_1(c(a)(w)), \pi_3(c(a)(w))) & \text{otherwise} \end{cases}$$

---

[4] The assignment $forget\_Sem$ is can be extended to a functor $forget\_Sem : CoAlg(G2) \to CoAlg(G)$, with behavior on coalgebra homomorphisms defined by $forget\_S(f) := f$.

Essentially, $forget\_Sem(incr2)$ considers the parsing behavior of $incr2$ only. The parser part of the transitions of $incr2$ are left intact, while labels denoting semantic increments are forgotten. Turning to the question of how to represent the relationship of $forget\_Sem(incr2)$ to $incr$, a bisimulation appears to be an appropriate property to capture this situation.

The notion of a bisimulation captures the notion of behavioral equivalence of two systems. One may obtain the definition of a bisimulation between coalgebras on a given Kripke polynomial functor F by defining for any relation $R \subseteq X \times Y$ the *lifted* relation $Rel(F)(R)$.

Definition 5.5.1 (Bisimulation of coalgebras)**.** A relation $R \subseteq A_1 \times A_2$ is now a bisimulation relation between F-coalgebras (A_1, c_1), (A_2, c_2) if and only if for all $a_1 \in A_1, a_2 \in A_2$:

$$R(a_1, a_2) \Rightarrow Rel(F)(R)(c_1(a_1), c_2(a_2))$$

Relation lifting is defined inductively on the structure of Kripke polynomial functors in Jacobs (2012)(p. 85), and is reproduced here.

Definition 5.5.2 (Relation lifting on Kripke polynomial functors)**.** If F is an endofunctor on sets, $R \subseteq Y \times Z$, define the lifted relation $Rel(F)(R) \subseteq F(Y) \times F(Z)$ by induction on the structure of F.

F is the identity functor

If F is the identity functor $X \mapsto X$, then:

$$Rel(F)(R) := R$$

Hence, if (A_1, c_1), (A_2, c_2) are F-coalgebras, R is a bisimulation if and only if for all $a_1 \in A_1, a_2 \in A_2$:

$$R(a_1, a_2) \Rightarrow R(c_1(a_1), c_2(a_2))$$

That is, we require that if two states of the coalgebras are related, the new states under the transitions of the respective coalgebras are also related.

F is the constant functor

If F is the constant functor $X \mapsto A$, then:

$$Rel(F)(R) := \{(a, a) : a \in A\} = \Delta_A$$

If (A_1, c_1), (A_2, c_2) are F-coalgebras, R is a bisimulation if and only if for all $a_1 \in A_1, a_2 \in A_2$:

$$R(a_1, a_2) \Rightarrow c_1(a_1) = c_2(a_2)$$

That is, related states are required to produce the same output-values.

F is the product functor

If F is a product functor $X \mapsto G(X) \times H(X)$, then:

$$Rel(F)(R) := \{((y', y''), (z', z'')) : Rel(G)(R)(y', z') \wedge Rel(H)(R)(y'', z'')\}$$

For instance, if G is the constant functor $X \mapsto B$, H the identity functor $X \mapsto X$. G coalgebras $(A_1, c_1), (A_2, c_2)$ are bisimilar if and only if for all $a_1 \in A_1, a_2 \in A_2$

$$R(a_1, a_2) \Rightarrow Rel(F)(R)(c_1(a_1) = (b, a_1'), c_2 = (b', a_2') \Leftrightarrow b = b' \wedge R(a_1', a_2')$$

That is, for related states we require that the output associated with each state in their coalgebra are identical, and the new states associated with each state in their coalgebra are related.

F is the set indexed sum functor

If F is a I indexed sum functor G_i, then:

$$Rel(F)(R) := \bigcup_{i \in I}(\{((u, i), (v, i)) : (u, v) \in Rel(G_i)(R)\})$$

For instance, consider if F is the sum $X \mapsto G + H$, and G is a constant functor, H the identity functor. Then coalgebras on F would be bisimilar if for all a_1, a_2 where $R(a_1, a_2)$ holds:

- $c_1(a_1) \in G(A_1)$ if and only if $c_2(a_2) \in G(A_2)$, in which case $c_1(a_1) = c_2(a_2)$. That is, a_1 is associated with an output only when a_2 is, in which case they are associated with the same output.

- $c_1(a_1) \in H(A_1)$ if and only if $c_2(a_2) \in H(A_2)$, in which case it is required that $R(c_1(a_1), c_2(a_2)$. That is, a_1 is associated with a new state only when a_2 is, in which case the two states are required to be related.

F is the powerset functor

If F is the powerset functor $X \mapsto \wp(G(X))$, then:

$$Rel(F)(R) := \{(U, V) : \forall u \in U(\exists v \in V(Rel(G)(R)(u, v))) \wedge \forall v \in V(\exists u \in U(Rel(G)(u, v)))\}$$

Consider the case when G is the product functor $X \mapsto (B \times X)$. Then, states of the respective coalgebras will be associated with sets of tuples. Consider for instance $(b, a_1') \in c_1(a_1)$. If $R(a_1, a_2)$ holds, bisimulation requires that there exists $(b, a_2') \in c_2(a_2)$, and that $R(a_1', a_2')$. Note however that nothing is stopping there from being $(b, a_2'') \in c_2(a_2)$ with $R(a_1', a_2'')$ but $a_2'' \neq a_2'$.

F is the exponent functor

If F is the exponent functor $X \mapsto I(X)^B$, then:

$$Rel(F)(R) \coloneqq \{(f,g) : \forall b \in B(Rel(I)(f(b), g(b)))\}$$

Consider the case where I is the sum functor $X \mapsto G + H$ from above. Then two coalgebras on the functor are bisimilar, if and only if for every pair of states a_1, a_2 where $R(a_1, a_2)$ we have that for all $b \in B$:

$$Rel(I)(R)(c_1(a_1)(b), c_2(a_2)(b))$$

That is, the conditions from the sum functor example are now imposed on the outputs of the respective functions that related states are associated with.

Getting back to the question of the bisimulation between $forget_S(incr2)$ and $incr$ (in the coalgebra G). The relation in question here is $R \subseteq (M \times P) \times P$ given by:

$$R \coloneqq \{((x,y),z) : (x,y) \in M \times P \wedge z \in P \wedge y = z\}$$

That is, the behavior of two states of the two coalgebras should be the same if they have the same parser state / developing syntactic representation. In order to decide if R is a bisimulation from $forget\_Sem(incr2)$ to $incr$, I define the lifted relation for the functor G, $Rel(G)(R)$, and require that for all $x \in M \times P, y \in P$,

$$R(x,y) \rightarrow Rel(G)(R)(forget\_S(incr2)(x), incr(y))$$

$Rel(G)(R)$ is given by:

$$\begin{aligned}&Rel(G)(R)(f,g) \Leftrightarrow\\&\forall w \in W((f(w) \in E \wedge g(w) \in E \wedge f(w) = g(w)) \vee (f(w) = (l,x) \wedge g(w) = (l,y) \wedge R(x,y)))\end{aligned}$$

But surely there are cases $(m,p) \in M \times P$, $p \in P$ where $forget\_S(incr2)(m,p)(w) \in E$ but $incr(p)(w) \notin E$. This is precisely the notion that $incr2$ will sometimes report a meaning-related error where $incr$ will not. Hence the bisimulation requirement is too strong.

Following Hughes and Jacobs (2004)(p. 77), a weaker condition, called a simulation, can be obtained by weakening the lifted relation. The definition involves defining an order $\sqsubseteq$ on the functor G (p. 73), which assigns to each set X a preorder $\sqsubseteq_X$ on $G(X)$.

The preorder on G will have the role of weakening $Rel(G)(R)$. For each set X define the preorder $\sqsubset_X \subseteq G(X) \times G(X)$:

$$\sqsubseteq_X (f,g) \Leftrightarrow \forall w \in W((f(w) \in E \wedge g(w) \notin E) \vee g(w) = f(w)) \Leftrightarrow \forall w \in W(P(f(w), g(w)))$$

$\sqsubseteq_X (f,g)$ holds when f is like g except on outputs of f in E where the corresponding output of g is not in E. $\sqsubseteq_X$ is a preorder as:

- It is reflexive: $f = g$ entails $f(w) = g(w)$ for all $w \in W$, and thus $\sqsubseteq_X (f, f)$ for all $f \in G(X)$.

- It is transitive. Assume $\sqsubseteq_X (f, g)$, $\sqsubseteq_X (g, h)$.
  For arbitrary $w \in W$, if $f(w) \in E$ and $g(w) \notin E$ then $h(w) \notin E$. Hence the condition P is satisfied.
  Alternatively, if $f(w) \in E$ and $g(w) \in E$ then $f(w) = g(w)$. If $h(w) \in E$ then $g(w) = h(w)$ and $f(w) = h(w)$ and the condition is satisfied. If $h(w) \notin E$ then the condition P is also satisfied.
  Finally, if $f(w) \notin E$ then $f(w) = g(w) \notin E$, $g(w) = h(w) \notin E$ and thus $f(w) = h(w)$, satisfying the condition P.
  Thus, $\sqsubseteq_X (f, h)$

Following Hughes and Jacobs (2004), a simulation is encoded in the following way. First, define an endofunctor on Rel, $\sqsubseteq_G \circ Rel(G)(-) \circ \sqsubset_G$ given for relations $R \subseteq X \times Y$ by:

$$R \mapsto \sqsubseteq_Y \circ Rel(G)(R) \circ \sqsubseteq_X$$

The requirement that R is a simulation from $forget\_Sem(incr2)$ to $incr$ is:

$$R(x, y) \Rightarrow \sqsubseteq_P \circ Rel(G)(R) \circ \sqsubseteq_{M \times P} (forget\_Sem(incr2)(x), incr(y))$$

That is, for all $(m, p) \in M \times P$, $p \in P$, we have that

$$\begin{aligned}
&\sqsubseteq_P \circ Rel(G)(R) \circ \sqsubseteq_{M \times P} (forget\_S(incr2)(m, p), incr(p)) \\
&\Leftrightarrow \exists x, y (\sqsubseteq_{M \times P} (forget\_Sem(incr2)(m, p), x) \wedge Rel(G)(R)(x, y) \wedge \sqsubseteq_P (y, incr(p)))
\end{aligned}$$

Let $x = forget\_Sem(incr2)(m, p)$, $y : W \to (E + (L \times P)$ be given by:

$$y(p)(w) := \begin{cases} forget\_Sem(incr2)(m, p)(w) & \text{if } incr(p)(w) \notin E \\ \quad \text{and for some for some } m \in M : forget\_Sem(incr2)(m, p)(w) \in E \\ incr(p)(w) & \text{otherwise} \end{cases}$$

Note that if $incr(p)(w) \notin E$ then $\forall m \in M (forget\_S(incr2)(m, p)(w) \in E \to forget\_Sem(incr2)(m, p)(w$ sem_err. Hence output in the first case of the definition above is uniquely defined.

By the reflexivity of preorders, we have that:

$$\sqsubseteq_{M \times P} (forget\_Sem(incr2)(m, p), forget\_Sem(incr2)(m, p))$$

Moreover, we have that $\sqsubseteq_P (y, incr)$. Proof:
Fix an arbitrary $p \in P$.

- If $y(p)(w) \in E$ and $incr(p)(w) \in E$ then $incr(p)(w) = y(p)(w)$ as the second case in the definition of y holds. This satisfies the condition P.
- If $y(p) \in E$ but $incr(p)(w) \notin E$ then the condition P is satisfied.
- If $y(p)(w) \notin E$ then $incr(p)(w) = y(p)(w)$ as the second condition of the definition of y holds, satisfying the condition P.

Finally, I prove that $Rel(G)(R)(x,y)$ holds, i.e. that $Rel(G)(R)(forget\_Sem(incr2),y)$. Fix arbitrary $(m,p) \in M \times P$, $p \in P$ ensuring that $R((m,p),p)$. Second, assume fixed but arbitrary $w \in W$.

- If $forget\_Sem(incr2)(m,p)(w) \in E$ then $forget\_S(incr2)(m,p)(w) = y(p)(w)$ by definition of y. Hence $Rel(X \mapsto E)(forget\_Sem(incr2)(w), y(w))$ holds.
- If $forget\_Sem(incr2)(w) = (syn,(m',p')) \in Syn\times(M\times P)$, then $y(p)(w) = incr(p)(w)$, by the second condition defining y. As $incr(p)(w) \in E \Rightarrow forget\_S(incr2) \in E$, $y(p)(w) = (syn',p'') \in Syn \times P$. As it was a requirement of $incr2$ that the addition of m would have no say in deciding on the new parser state p, $p' = p''$. Hence $R((m',p'),p'')$ as required by $Rel(X \mapsto Syn \times X)(R)$. Finally, as $incr2$ inherits labels from $incr$, $syn = syn'$, as required by $Rel(X \mapsto Syn \times X)(R)$.

Thus for all $(m,p) \in M \times P$, $p' \in P$:

$$R((m,p),p') \Rightarrow \sqsubseteq_P \circ Rel(G)(R) \circ \sqsubseteq_{M\times P}$$

Proving that R defines a simulation from $(M \times P, forget\_Sem(incr2))$ to $(P, incr)$.

5.6 Updated reanalysis algorithm

In this section, I update the reanalysis algorithm with meanig representations.

The original functor on which the state space was defined was J given by.

$$X \mapsto E + (\wp((0,1) \times Syn \times X))^W$$

For the updated reanalysis algorithm, I will use a functor $J2$:

$$X \mapsto E + (\wp((0,1) \times Sem \times (0,1) \times Syn \times X))^W$$

Together with a coalgebra with $(M \times P, incr2\_alt)$. As with $incr\_alt$, $incr2\_alt$ is subject to a monotonicity requirement, and an injectivity requirement. Recall the substructure relation $sub_{M\times P}$. The monotonicity requirement on $incr2\_alt$ is that for all $(m,p) \in M \times P$, for all

$w \in W$:

$$(prob_{sem}, sem, prob_{syn}, syn, m', p') \in incr2\_alt(m,p)(w) \Rightarrow \leq_{M \times P} ((m,p),(m',p'))$$

The injectivity property is now:

$incr2\_alt(m,p)(w) \in \wp((0,1) \times S \times (0,1) \times L \times X)$
$\Rightarrow < r_2, r_4 >: R \subseteq ((0,1) \times S \times (0,1) \times L \times X)) \times ((0,1) \times S \times (0,1) \times L \times X)) \to S \times L$ is injective

Where r_i gives the i-th coordinate of a tuple, $< r_2, r_4 >$ is the function assigning to each tuple the 2-tuple of the second and fourth coordinates. The injectivity property essentially states that alternatives are identifiable by the combination of syntactic and semantic transition labels involved.

However, care must be taken when adopting such a coalgebra, or the simulation property from the previous section will be broken when $incr2\_max$ is defined. If implausibility considerations are used to remove syntactic alternatives from $incr\_alt$ when $incr2\_alt$ is defined, this may cause $incr2\_max$ to make different syntactic decisions than $incr\_max$. Critically, if for some $(m,p) \in M \times P$, for some w, $incr2\_alt(m,p)(w)$ does not contain the highest ranked option from $incr\_alt(p)(w)$, $incr\_max$ will make different syntactic increments than $incr$ as the next-to-highest option is chosen.

Ensuring that all syntactic options are kept can be expressed as a bisimulation requirement. First define J-coalgebra $forget\_Sem(incr2\_alt)$ defined by:

$$forget\_Sem(incr2\_alt)(m,p)(w) :=$$
$$\begin{cases} incr2\_alt(m,p)(w) & \text{if } incr2\_alt \in E \\ \{(prob\_syn, syn, m, p) : (prob\_s, s, prob\_l, l, m, p) \in incr2\_alt(m,p)(w)\} & \text{otherwise} \end{cases}$$

The relation in question here is again $R \subseteq (M \times P) \times P$ given by:

$$R := \{((x,y),z) : (x,y) \in M \times P \wedge z \in P \wedge y = z\}$$

In order for R to be a bisimulation between J-coalgebras $forget\_Sem(incr2\_alt)$ and $incr\_alt$ it is required that:

$$R((m,p),p) \Rightarrow Rel(HR)(R)(forget\_Sem(incr2\_alt)(m,p), incr\_alt(p))$$

Following the definition of relation lifting, $Rel(J)(R)(f,g)$ holds if and only if for all $w \in W$:

- Either $f(w) = g(w) \in E$
- Or

$$
\begin{aligned}
&\forall x \in f(w) \in W(\exists y \in g(w)(Rel(X \mapsto (0,1) \times L \times X)(x,y))) \wedge \\
&\forall x \in g(w) \in W(\exists y \in g(w)(Rel(X \mapsto (0,1) \times L \times X)(x,y)))
\end{aligned}
$$

That is, whenever $forget\_Sem(incr2\_alt)(m,p)(w)$ ends up in $(0,1)\times L\times M\times P$, $incr\_alt(p)$ should end up in $(0,1) \times L \times P$, and only then. Moreover, for each

$$(prob\_syn, syn, (m', p')) \in forget\_Sem(incr2\_alt)(m,p)(w)$$

there should exist:

$$(prob\_syn, syn, p) \in incr\_(alt)(p)(w)$$

Additionally, for each

$$(prob\_syn, syn, p) \in incr\_(alt)(p)(w)$$

there should exists

$$(prob\_syn, syn, (m', p')) \in forget\_Sem(incr2\_alt)(m,p)(w)$$

A single syntactic alternative for $incr\_alt$ will often correspond to multiple syntactic alternatives of $forget\_Sem(incr2)$, owing to different ways of integrating output semantically. This does not break the bisimulation property.

As discussed in the previous section, $incr2$ should return an error where $incr$ manages fine. That is, semantic properties of a developing representation should some times cause backtracking. However, if synactic options are not eliminated in the state space, $incr2\_alt$, how can they be eliminated? I have opted for the introduction of a special semantic transition symbol $\perp$, with which to tag syntactic options with no semantic continuations.

The meaning of $\perp$ will become clear when $incr2\_max$ is defined. Recall that $(P, incr\_max)$ was a coalgebra on the functor:

$$X \mapsto (E + (\wp((0,1) \times L \times X) \times L \times X))^W$$

$incr2\_max$ will however be defined on a functor L, given by:

$$X \mapsto (((E\times\wp((0,1)\times Sem\times(0,1)Syn\times X))+(\wp((0,1)\times Sem\times(0,1)\times Syn\times X)\times Sem\times Syn\times X))^W$$

Note that coalgebras on this functor are required to couple a set of alternatives with errors (possibly the empty set). Errors will be coupled with a non-empty set when the highest ranked syntactic alternative is tagged with $\perp$.

Define $K2$-coalgebra $(M \times P, incr2\_max)$ defined for $(m,p) \in M \times P$ by the assignment:

$$
\begin{aligned}
&incr2\_max(m,p)(w) := \\
&\begin{cases}
(err, \varnothing) \in E \times \wp((0,1) \times S \times (0,1) \times L \times M \times P) \text{ if } incr\_alt(p)(w) = err \in E \\
(sem\_err, Alts') \in E & \text{if } s = \bot \\
\text{ where } \exists (prob_{sem}, sem, prob_{syn}, syn, m', p') \in incr2\_alt(p)(w)\big(\\
\quad \forall (x_1, x_2, x_3, x_4, x_5, x_6) \in incr2\_alt(p)(w)(prob_{syn} \geq x_3) \quad \wedge \\
\quad \forall (y_1, y_2, prob_{syn}, syn, y_5, p') \in incr2\_alt(p)(w)(prob_{sem} \geq y_1)\big) \\
\quad Alts := incr\_alt(p)(w) \smallsetminus \{(r, s, r', l, m', p')\}\big) \\
\quad Alts' := \{(z_1, z_2, z_3, z_4, z_5, z_6) \in Alts : z_2 \neq \bot\} \\
(Alts', (l, s', m', p')) & \text{otherwise}
\end{cases}
\end{aligned}
$$

The behavior of $incr2\_max$ is the following. If the underlying transition space $incr2\_alt$ returns an error, return the same error (with no alternatives). If the highest ranked semantic alternative is tagged with $\bot$, return an error, together with every other alternative, except for alternatives tagged with $\bot$. It is important to remove every alternative tagged with $\bot$, as the reanalysis algorithm will attempt to restart $incr2\_max$ in the alternatives. In the present model, probabilities of synactic options dominate probabilities of semantic options. Alternative definitions of $incr2\_max$ that maximize combinations of syntactic and semantic probabilities are possible, but such algorithms move in the direction of constraint based parsing. That is, properties of meaning representations would be able to influence syntactic decisions.

The updated reanalysis algorithm can now be presented. Let $L2$ be an endofunctor on set, given by. Note that compared to the corresponding functor (L) in 4.3, that the list of labels is now a list of pairs of semantic and syntactic labels.

$$X \mapsto (E + ((Sem \times Syn)^* \times X))^{(W^*)}$$

Define $L2$-coalgebra $incr\_rean$:

$$
\begin{aligned}
&incr2\_rean : W^* \times \mathbb{N} \times \wp((0,1) \times M \times (0,1) \times P \times \mathbb{N} \times (Sem \times Syn)^*) \times (Sem \times Syn)^* \times M \times P \\
&\to (E + ((Sem \times Syn)^* \times (W^* \times \mathbb{N} \times \wp((0,1) \times M \times (0,1) \times P \times \mathbb{N} \times (Sem \times Syn)^*) \times (Sem \times Syn)^* \times P))^{W^*}
\end{aligned}
$$

$$\mathit{incr2\_rean}(\mathit{all}, n, \mathit{AltNL}, \mathit{labels}, (m, p))(\mathit{nil}) := (\mathit{labels}, \mathit{all}, n, \mathit{AltNL}, \mathit{labels}, (m, p))$$

$$\mathit{incr2\_rean}(\mathit{all}, n, \mathit{AltNL}, \mathit{labels}, (m, p))(\mathit{ws}) :=$$

$$\begin{cases}
\mathit{incr2\_rean}(\mathit{all}, n+1, \mathit{AltNL}', \mathit{conc}(\mathit{labels}, [(\mathit{sem}, \mathit{syn})]), p')(\mathit{rest}(\mathit{ws})) \\
\quad \text{where } \mathit{AltNL}' := \begin{cases} \mathit{AltNL} \cup (\mathit{Alt} \times \{n\} \times \mathit{labels}) & \text{if } \mathit{head}(\mathit{ws}) \neq \text{"."} \\ (\mathit{Alt} \times \{n\} \times \mathit{labels}) & \text{otherwise} \end{cases} \\
\quad \text{if } \mathit{incr2\_max}(m, p)(\mathit{head}(\mathit{ws})) = (\mathit{Alt}, (\mathit{sem}, \mathit{syn}), (m', p')) \\
\\
\mathit{incr2\_rean}(\mathit{all}, n', \mathit{AltNL}'', \mathit{conc}(\mathit{labels}, [(\mathit{syn}, \mathit{sem})]), p')(\mathit{listfrom}(\mathit{all}, n')) \\
\quad \text{where } \mathit{AltNL}' := \mathit{AltNL} \cup (\mathit{Alt}' \times \{n\} \times \mathit{labels}) \\
\quad \mathit{choose2}(\mathit{AltNL}) := (\mathit{AltNL}'', ((\mathit{syn}, \mathit{sem}), (m', p'), n', \mathit{labels}))) \\
\quad \text{if } \mathit{incr2\_max}(m, p)(\mathit{head}(\mathit{ws})) = (x, \mathit{Alt}') \in E \wedge \mathit{AltNL}' \neq \varnothing \\
\\
\mathit{incr2\_max}(m, p)(\mathit{head}(\mathit{ws})) \in E \\
\quad \text{if } \mathit{incr2\_max}(m, p)(\mathit{head}(\mathit{ws})) = (x, \varnothing) \in (E \times \wp((0,1) \times S \times \ldots) \wedge \mathit{AltNL} = \varnothing
\end{cases}$$

The function $\mathit{choose2}$ selects a single alternative from a set of alternatives, returns the updated set of alternatives and the chosen alternative.

$$\mathit{choose2} : \wp((0,1) \times M \times (0,1) \times P \times \mathbb{N} \times (\mathit{Sem} \times \mathit{Syn})^*) \rightarrow$$
$$\wp((0,1) \times M \times (0,1) \times P \times \mathbb{N} \times (\mathit{Sem} \times \mathit{Syn})^*) \times (\mathit{Sem} \times \mathit{Syn} \times M \times P \times \mathbb{N} \times (\mathit{Sem} \times \mathit{Syn})^*$$

I assume that $\mathit{choose2}$ first selects the alternative with the highest syntactic ranking, and then selects the alternative with the highest semantic ranking from these.

Note that at present, the algorithm does not discriminate between reprocessing caused by semantic error and reanalysis. If there is a strictly syncatic error, the processor might choose a different semantic reading with the same syntactic structure. An additional problem with the algorithm is the fact that it is the underlying mechanism $\mathit{incr2\_max}$ that decides when reprocessing happens. This problem was in fact inherited from the original algorithm in 4.3. Semantic errors are presumably a matter of degree, and sensitivity to such errors is something one would like to adjust in different situations. In the present algorithm, it is impossible (without a major redesign) to adjust the sensitivity for reprocessing in the control structure. Instead, changes have to be made in the underlying state space. In experimentation with parameters, it might be better to have the option to adjust this sensitivity in the reprocessing algorithm itself.

Psycholinguistically speaking, we know little about how reanalysis happens to begin with, and even less about how reanalysis differs when it is caused by semantic errors compared to when it is caused by syntactic errors. The algorithm proposed above is naïve. However, interesting questions are brought up by considering such an algorithm. For instance, how does $choose2$ behave? Does the dominance of syntax extend to reprocessing, or can semantic errors cause semantic factors to be given priority when choosing an alternative starting point?

Chapter 6

Extending the model with states of knowledge activation

The present chapter is about adding states of knowledge activations to the model. The notion of a state of activated knowledge guiding further processing is proposed both in the construction integration model proposed by Kintsch (1988) and in the structure building framework elaborated in Gernsbacher (1990).

Such states should encode an expectancy of the *topics* or *areas of knowledge* that a text is about, which in turn will advantageously guide language processing. Kintsch (1988) summarises such an idea:

> In a word, knowledge makes understanding processes smart: It keeps them on the right track and avoids exploring blind alleys. People understand correctly because they sort of know what is going to come. (p. 164).

I argue that mental states that represent such topic level-context influence incremental processing at a very early stage. The early influence of states of activated knowledge can help ameliorate the problem of computational tractability when coverage is extended to new domains, improve the accuracy of the parser, and processing some types of metaphor.

The first section of the chapter is about what kinds of meaning representations are actually produced when people understand language (6.1). These are coded in terms of conceptual knowledge about the world. Research has shown that the interface between language and conceptual knowledge is direct, with little evidence of intermediary meaning representations.

The next section of the chapter is about evidence linking activations of knowledge structures to incremental processing. Given these findings, I extend the model with states of knowledge activation. In the subsequent sections, I investigate the use of extending the model with states of knowledge activation in solving the problems outlined in 2. I argue that the extended model can reduce problems with computational tractability associated with the standard pipeline by resolving word homonym level word-senses very early. There is also preliminary evidence that states of knowledge activation can inform incremental parsing decisions (6.5), improving parsing accuracy.

I argue that states of knowledge activation cause a reranking of alternative incremental decisions in the state space underlying the model. Some times, this reranking causes the default incremental decisions to change, while other times, the reranking is not felt until the decision is revisited in reprocessing.

When we read words, we resolve broad aspects of word meanings quickly (6.4.1). Many word meanings are often accessed, but more frequently used meanings are accessed before less frequently used meanings. Context causes meaning selection or causes infrequent meanings to be accessed earlier, so they compete with frequently used meanings for selection. There are also results indicating that global context specifically, such as topic information, causes meaning selection and meaning competition. NLP research has found that domain vectors, representing the differential involvement of domains of discourse in text, perform well in when distinguishing broad word meanings. This means they are good candidates for states of activation of knowledge in the model.

The results on early resolution of meanings by context only extend to broad meaning distinctions, and not to fine nuances in word meaning. Finer nuances in meaning are left undecided when a word is processed. Finer aspects of word meaning then, likely depend more on sentence-local context, and should be processed in a different way than broad aspects of word meaning. In NLP research, a word will typically only occur in one broad meaning in a text, while words can occur with several finely distinct meanings in a text, indicating that the topic of a text is not sufficient to distinguish such meanings. Correspondingly, I argue that transitions should have labels denoting only broad meaning distinctions.

Next, I discuss preliminary research indicating that states of knowledge activation may be useful in determining incremental syntactic decisions, and that such an influence does not necessarily involve a constraint based view of parsing.

6.1 Elementary findings on meaning representations in language comprehension

The following section summarizes evidence on the meaning representations that result from language comprehension. The purpose of the present section is to argue that the meaning representations that result when comprehending discourse are built from our knowledge of the world. Moreover, I argue that language is comprehended in tight interaction with knowledge.

6.1.1 The product of comprehension is situation models

The classical approach to the problem of formalizing natural language expressions automatically is Montague semantics (Janssen, 2012). In short, montague semantics provides an automatic way of assigning logical formulae to a fragment of English. These logical formulae encode the truth conditions of an utterance, and are then used in logical inference. Psycholinguistic research however has questioned the role of such representations in human language processing.

Seminal research by Bransford, Barclay, and Franks (1972) provided evidence that comprehension processes lead to representations of the situations described by language. Bransford et al. argued that representations that uniquely determine the interpretations of sentences, do not correspond to what is actually remembered after comprehending a sentence. I.e., they argue that comprehenders do not construct fully disambiguated representations of sentence meaning up to or parametrized by the interpretation of the involved relations and terms. Rather, it appears that comprehenders instantiate a model of the situation being described, built from conceptual knowledge structures.

Bransford et al. (1972) presented participants with sentences such as either 1. and 2. below. Note that in sentence 1 it follows that the dogs circled around the tree.

1. The raccoons raced up the tree and the dogs circled around them.
2. The raccoons looked over towards the tree and the dogs circled around them. (p. 197)

The presentations were followed a few minutes later by a recognition test that included some new sentences. Participants were presented the original sentences and altered sentences such as either 3. and 4. below. Note that sentence 3 makes the inference in 1 explicit. Participants were asked to indicate whether they had been presented the sentence earlier and their confidence that the sentence had been presented.

3. The raccoons raced up the tree and the dogs circled around it.
4. The raccoons looked over towards the tree and the dogs circled around it. (p. 197)

For a given pair of sentences, some participants were presented the sentence with the possible inference (1) in the first part of the experiment. Of these participants, some were presented sentence 1 in the recognition test, while others were presented 3. There were no statistically significant differences in recognition of 1 and 3 among participants having been presented 1. When presented sentence 2, participants could distinguish between the original sentence and the modified version (2 and 4) in the recognition test.

What was remembered then, was likely a representation of situation described by a sentence, and not an unambiguous representation of the truth conditions of the sentence. Indeed, the final representation appears to be much richer than the strict declarative content of the sentence. The comprehenders appear to have created an internal representation of the events that were described, where assumptions derived from world knowledge were made effortlessly.

In modern psycholinguistics, many researchers agree that the meaning representations that comprehenders produce when reading coherent discourse thought of as *situation models* [1]

---

[1] For a review, see Zwaan (1999) or Radvansky and Dijkstra (2007).

Early contributions to the theory of situation models include van Dijk and Kintsch (1983) and Johnson-Laird (1983). Situation models are thought to represent the situation described by language, in contrast to the earlier focus on representing the truth conditions of utterances unambiguously. Situation models can be seen instantiations of generalized conceptual knowledge. Although the evidence outlined in this section does not constitute evidence on incrementality, the findings do tell us about which psychological constructs should correspond to formal representations if language processing is incremental.

6.1.2 Semantic memory encodes situation models

Situation models involve a type of memory called *semantic memory*, which is thought to be heavily involved in language comprehension. Semantic memory denotes “knowledge about people, objects, actions, relations, self, and culture acquired through experience” (J. R. Binder, Desai, Graves, & Conant, 2009, p. 2767). An in-depth discussion on findings on semantic memory is outside the scope of the present text. The focus in the text is on how language is comprehended. A relevant question then is when semantic memory is brought to bear on language processing. The present section argues situation models are encoded in terms of semantic memory, and that the correct memory structures must be accessed when the sentence is being processed.

Bransford and Johnson (1972) found evidence that it is necessary to access the correct structures in memory for effective language comprehension. In experiments II and III, they presented participants with a passage of text that described the notion of washing clothes in a rather abstract fashion, such that one could not with ease see that this was the topic of the passage without being informed of it. The passage contained sentences such as:

> The proceduce is actually quite simple. First you arrange things into different groups depending on their makeup. Of course, one pile may be sufficient depending on how much there is to do. (p. 722).

Participants who learned about the topic of the text after reading it, or did not learn about the topic at all both performed much worse on measures of comprehension and recall than those who were told what the the topic of the passage would be in advance. Evidently, linking language to semantic memory is necessary for humans to comprehend language, and it is important that we know what utterances are about when we process them. This result may occur because comprehenders are unable to store intermediary representations, as suggested by J. A. Fodor (1983, Chapter 3, part III), or it may be because access to such knowledge is critical at some earlier stage in the comprehension process.

Further evidence is provided by research investigating the effects of relevant knowledge on language comprehension. According to the theory of situation models, humans should comprehend language with greater ease the more knowledge they have about the situations underlying a story or description. W. Schneider, Körkel, and Weinert (1989) presented a story about a soccer player to children with different levels of knowledge about soccer. The story could be understood by listeners with little knowledge of soccer. The study found that children with low scores on the verbal component of an aptitude test, but with more extensive knowledge about soccer outperformed higher scorers on the verbal aptitude test (with less knowledge of soccer) when tested on their memory and comprehension of the story. Presumably, the children with more knowledge of soccer could more readily build and reason about the situation models corresponding to the story. Similar findings on the positive effects of domain knowledge on language comprehension have been made by Spilich, Vesonder, Chiesi, and Voss (1979) and Hambrick and Engle (2002).

6.2 Evidence for the early influence states of knowledge

In the present section, I argue that states of activation of knowledge about the world interface with language comprehension at an early stage. This section forms the psycholinguistic motivation for an extension of the model in the next section. Evidence that this is the case comes from the N400 effect.

6.2.1 The lexical view of the N400 effect

The lexical view of the N400 effect states that the effect is caused by "... facilitated activation of features of the long-term memory representation that is associated with a lexical item" (Lau et al., 2008, p.921). That is, the difference of N400 amplitudes at a critical word in control vs. manipulation sentences is interpreted as differing availability of the memory representation associated with the critical word. This view implies that words interface with knowledge structures at an early stage of processing, as the N400 effect is explained by the process of *accessing* conceptual knowledge structures.

There are findings that the N400 corresponds to activity in the anterior temporal lobe (Federmeier & Kutas, 1999, p. 471), an area believed to be highly important in conceptual knowledge. Damage to the anterior temporal lobe is associated with impaired conceptual knowledge across domains (Rogers et al., 2006). Kutas and Federmeier (2000, p. 465) note that the N400 is sensitive to manipulations of word frequency. Moreover, if the lexical view of the N400 holds, the N400 should also be sensitive to manipulations of the state-level accessibility of knowledge structures.

Federmeier and Kutas (1999) looked at N400 responses associated with words in a mismatching context. For instance, in a context where participants expected the word "tulip" (assessed through a sentence completion procedure), Federmeier and Kutas compared N400 responses when participants were instead presented "roses" and "pines". If the concept of a tulip is already activated, it should be easier to access the related concept of a rose than the more distantly concept of pine. If the N400 effect is caused by differences in conceptual access as the lexical hypothesis claims, the difference in the access process should be reflected in N400 differences aswell. Consider the following experimental material from Federmeier and Kutas (1999):

> The tourist in Holland stared in awe at the rows and rows of color. She wished she lived in a place where they grew *tulips/roses/pines*. (p. 473, original emphasis)

Here “tulips” are expected, “roses” mismatch, but are of the same category, and “pines” mismatch and belong to a different category. As expected, N400 amplitudes were lower for mismatching target words of the same category than for those in a different category.

In recent research using the N400, Metusalem et al. (2012) found that activated *event* knowledge influences language processing at an early stage even when abstracting away low level semantic associations. In the first experiment, participants were presented with three different descriptions of typical scenarios: descriptions including expected words, event-related but unexpected words and event-unrelated unexpected words respectively. The main finding was that the event-related, unexpected words elicited N400 responses with lesser amplitudes than event-unrelated unexpected words. This association remained even when cooccurrence of event-related words with the preceding context was controlled for, meaning the effect is not likely a result of mere cooccurence. According to Metusalem et al. (2012), this result indicates that activation of event related knowledge structures influences and indeed interacts with on-line sentence processing at an early stage.

The lexical view of the N400 and the evidence that supports it indicates that there is an aspect to the mental state of the comprehender that keeps track of what knowledge structures are relevant to upcoming words and sentences in coherent discourse. This state influences language processing at an early stage. In the next section, I will update the model in a corresponding manner.

6.3 Extending the model with states of knowledge activation

A set A will be assumed to index states of activation of knowledge structures in semantic memory. In the extended model, states now will not only reflect combinations of parser states and meaning representations, but also states of activation of knowledge. I.e. the new carrier will be $A \times M \times P$.

At times, there will be an overlap in the information contained in M and in A. That is, one might derive information about states of activated knowledge from developing meaning representations. There is however a good reason to keep the two representations seperate. The information contained in M only spans a single sentence. Activations of knowledge however, span sentence boundaries. For instance, we would like to be able to hold an expectancy of what a sentence will be about before processing a single word. It should also be possible for knowledge structures activated by one sentence to influence how the next sentence is processed.

Let A be a set of all knowledge activations. Suggestions as to the contents of A will be made in subsection 6.4. The new carrier is $A \times M \times P$, with a G-coalgebra $incr3$:

$$incr3 : A \times M \times P \to (E + (Sem \times Syn \times (A \times M \times P)))^W$$

With M and P it was required that there was a distinguished representation $empty$ in each set in order to represent initial configurations when processing a new sentence. Note however that no such requirement is put on A. There might however be a default state $default$ or $baseline$, which does not necessarily afford each knowledge structure the same activation.

A surprising finding in natural language processing is that language is brimming with ambiguities that language comprehenders are simply not aware of (cf. Jurafsky & Martin, 2009, pp. 466-468). These ambiguities it is said, must be resolved using knowledge. The standard pipeline defers the resolution of ambiguities with knowledge until after they are produced. The primary role for A in the carrier is to attempt to draw in knowledge at an earlier stage, so as to nip many ambiguities in the bud, before they multiply throughout processing.

In the following sections, I argue that the states of knowledge activation just added to the model can indeed to resolve senses at an early stage of processing and improve the accuracy of parsing.

6.4 Use of states of knowledge activation in resolving word senses

The NLP task of word-sense disambiguation (WSD) has considerable overlap with the process of lexical ambiguity resolution which is postulated in psycholinguistics, although WSD might involve finer grained semantic distinctions. In the present section, I argue that states of knowledge activation are useful in disambiguating senses at a very early stage, but that fine grained distinctions are postponed.

First, I discuss elementary research on mechanisms of lexical ambiguity resolution by context (6.4.1).

Then, I discuss evidence on the kinds of sense distinctions that are encoded lexically, and thus the kinds of lexical distinctions that are made by context (6.4.3). There is considerable evidence that fine sense distinctions are not done incrementally, but are deferred until a stage.

Still missing however is evidence that mental representations of the topic or domain of knowledge in a text (corresponding to knowledge activations) mediate a practically significant proportion of the influence of context on lexical sense disambiguation. Such evidence is discussed in 6.4.4.

Finally, I summarize the implications of the findings discussed in the present section for the model (6.4.5).

6.4.1 Mechanisms of lexical ambiguity resolution

There is agreement that the context influences the resolution of lexical ambiguities, and that such effects happen during incremental processing (Morris, 2006, p. 386). The disagreement has centered around at which point in the process the contextual influences play a role. One possible mechanism is that context acts on retrieval processes, ensuring that meanings / lexical items are *selectively accessed / activated* according to their relevance to context. Another possibility is that all meanings are *exhaustively* accessed, and that context helps select the relevant lexical item (and to exclude irrelevant lexical items) (Morris, 2006, p. 382).

There is evidence that in the absence of a disambiguating context, multiple meanings of ambiguous words are accessed, supporting an exhaustive access model (Morris, 2006, pp. 381-382). Consider for instance the findings of (Seidenberg, Tanenhaus, Leiman, & Bienkowski, 1982). Participants were presented clauses with ambigous or unambiguous final words, followed by a target word. The Stimulus Onset Asynchrony (SOA) for target words was either 0ms after the presentation of the final word, or 200ms after presentation of the final word. The time taken to begin to read the target word was monitored.

(1) a. If Joe buys the straw - SIP
    b. If Joe buys the straw - HAY (p. 498)

In (1), the word “straw” is ambiguous between a plant/farming related reading and a drinking straw reading. If participants access only one of these meanings of straw initially, they should have facilitated access to only one of “SIP” or “HAY” when it is presented immediately after “straw”. If both meanings are accessed, facilitation should result for both “SIP” and “HAY”.

(2) a. If Joe buys the wheat - HAY
    b. If Joe buys the soda - SIP (p. 498).

In (2), the final words are unambiguous, and the target words are related to this unambiguous meaning. Thus we expect immediate facilitation of both target words for both multiple (exhaustive) access and selective access hypotheses. Compared to (1) then, predictions of selective access and multiple access differ. If the selective access hypothesis is true, target word response times should be higher for the ambiguous clauses (1) than for the unambiguous related clauses (2) regardless of SOA. This is because selective access causes facilitated access to the target word only when the *relevant* meaning is selectively accessed in (1), whereas the relevant meaning is always accessed in (2).

Moreover, if the multiple access hypothesis holds, then after an increased SOA (200ms), participants should have selected only one sense of “straw” in the ambiguous condition (1). Thus, they should on average take longer to read the target word, as the meaning they have selected will only be relevant part of the time. The findings of Seidenberg et al. were in line with the multiple access hypothesis. Participants spent as much time reading target words when

SOA was immediate, but more time reading the target words in the ambiguous condition (1) than in the unambiguous condition when SOA was delayed. With a delayed SOA, participants appeared to have selected one meaning of "straw", and spent on average more time reading the target, as it would some times be related to the non-chosen meaning of "straw".

However, in experiment 2, Seidenberg et al. (1982) found that when provided context biasing interpretation towards one of the possible meanings, such as the clause:

(3) a. Although the farmer bought the straw - HAY
    b. Although the farmer bought the straw - SIP

Response times to target words were affected immediately, suggesting that one meaning was indeed selectively accessed. In terms of the model then, such an influence of "farmer" is encoded by activation of related knowledge structures, ensuring selective access to one meaning of "straw".

Other research has found that context effects are in general not sufficient to result in selective access. Often, the senses of a word will have lopsided frequencies, such words are called *biased* (ambiguous) words. More frequent senses of biased words are called *dominant* senses, wheras less frequent senses are the *subordinate* senses. Without contextual manipulation, dominant senses tend to be accessed earlier than subordinate senses (Morris, 2006, p. 382). For instance, Simpson and Burgess (1985) conducted a study using a paradigm similar toSeidenberg et al. (1982). In experiments 1 and 2, Simpson and Burgess presented participants with biased ambiguous words (primes), followed by a target nonword or a word related to either the dominant or the subordinate sense of the target word (pp. 29-30). They were asked to decide if the target stimuli was a word or a nonword. The SOA between prime and word/nonword was varied. For words relating to the dominant meaning of the prime, performance (reaction times) on lexical decisions was facilitated immediately. For words relating to the subordinate meaning of the prime, facilitation in the lexical decision task was delayed (pp. 30-32). The results indicate that access to word senses is ordered, depending on dominance.

Contextual influences are not in general sufficient to ensure selective access to subordinate senses. It appears that dominant senses are many times accessed regardless of context (K. S. Binder & Rayner, 1998). Moreover, it seems that subordinate senses are some times not accessed when contexts do not prime them. The *subordinate bias effect* is the finding that participants take longer to read a biased word when the context is related to a subordinate meaning than when it is not (Duffy, Morris, and Rayner, 1988; Morris, 2006, p. 383). For instance, when context primes the wedding-related, subordinate meaning of "band", reading times for "band" are delayed compared to when context primes the dominant meaning of "band" up to baseline differences of "band" reading times (Kambe, Rayner, & Duffy, 2001)[2]. The subordinate bias effect is regarded as robust, as it has since been replicated in numerous studies (cf. Sereno,

---

[2] Of course, more experimental materials than just the one involving the word "band" were included in Kambe et al. (2001).

O'Donnell, & Rayner, 2006, p. 336).

In terms of selective and multiple access, the subordinate bias effect appears not to be wholly compatible with either hypothesis. Indeed, it appears that the mechanisms of lexical ambiguity resolution involving selective access and later selection (as in exhaustive access models) are possible. Neither are sufficient to explain the subordinate bias effect. In the *reordered access model*, structure is added to the access mechanism in the exhaustive access model. Instead of accessing all meanings at the same time, meanings are accessed in order. The order in which meanings are accessed is determined both by a static variable of relative sense frequency, and by contextual activation of certain meanings (Duffy et al., 1988, p.440-442).

As in exhaustive access, there are according to Duffy et al. (1988, p.441) processes that integrate a lexical item (presumably with sentence-level representations). This process has a significant role to play in lexical ambiguity resolution, but is not the main focus of the reordered access model.

For instance, Gernsbacher, Varner, and Faust (1990, Experiment 4) found that less skilled readers (evaluated by a text comprehension test) were poorer at suppressing senses of words inappropriate to a context, compared to more skilled readers. Approximately one second after reading sentences like "He dug with a spade", containing the homonym "spade", the less skilled readers had poorer performance (measured in reaction times) when deciding whether an unrelated word "ace" was related to the meaning of the sentence or not. The performance of more skilled and less skilled readers did not differ if the target word was given immediately after the prime sentence. In terms of the reordered access model, it appears that the context was not strong enough to reorder access to the contextually inappropriate sense of "spade". Similar findings have also been obtained by (van der Schoot, Vasbinder, Horsley, Reijntjes, & van Lieshout, 2009).

There is however evidence that strong contexts sometimes select senses (e.g. (Colbert-Getz & Cook, 2013)). Consider the case where a context reorders access to the senses to such a degree that there is no time for an alternative sense to be considered. In this case, it is unclear which empirical predictions of the reordered access model distinguish it from the predictions of a selective access model. Possibly, alternative senses of a word are accessed if reprocessing occurs at a later stage.

In conclusion then, lexical disambiguation occurs incrementally, soon after comprehenders read a word. Seidenberg et al. (1982) found that participants likely had selected one of the senses of "straw" (although somewhat delayed), even when no preceding disambiguating context was availabile. As readers commit to one lexical identity during incremental processing, it is appropriate to label transitions with lexical identities.

Multiple meanings are some times accessed when comprehenders read a word. However, access to meanings is ordered, meaning dominant meanings are accessed before subordinate meanings. According to the reordered access model, context can cause reordered access to the meanings of a word. When meanings are balanced in frequency, or a meaning is already dominant, context can cause selective activation of the meaning. When a meaning is subordinate, context is not in general sufficient to cause selective activation, but causes the subordinate sense to be accessed at an earlier stage, causing competition with the dominant sense.

6.4.2 States of knowledge activation in lexical disambiguation

The notion that states of knowledge activation should be useful in word sense disambiguation appears likely on first glance. Coarse grained word senses are encoded by differing lexical identities, and these are tightly connected to conceptual knowledge. No wonder then if activating knowledge structures relating to lexemes translates into selection of- or reordering of access to that lexeme. The present section further corroborate this notion with more direct evidence from coherent text. Activations of knowledge have a role as preprocessing stage, at least for coarse grained sense distinctions.

There is evidence that knowing the topic of a text acts as a preprocessing stage when comprehending language. According to the structure building framework of Gernsbacher (1990), comprehenders must initially lay the foundations of meaning representations when they comprehend text[3]. An important part of this process is activating the correct memory structures. Several studies have found positive effects of topic headings on comprehension of subsequent text (c.f. Hyönä & Lorch, 2004; Kambe et al., 2001).

If states of knowledge activation aid comprehension by resolving senses, as is proposed above, the positive effects of topic headings should be found specifically for ambiguous words in text. Wiley and Rayner (2000) found direct evidence that this is the case. In experiment 1, they presented participants with texts containing ambiguous words, among these the text from Bransford and Johnson (1972) reproduced in 6.1.2. When the texts had a descriptive title, participants read the nouns in the text faster (as indexed by shorter fixation times) and spent less time reprocessing the sentence than when no title was presented.

Stronger evidence emerged from experiments 2 and 3. Subordinate senses differ in frequency of use relative to their dominant counterparts (as do dominant senses to subordinate counterparts). This means that the dominant sense of one word can be more dominant than the dominant sense of another word, and that the subordinate sense of one word can be more subordinate than the subordinate sense of another word. Experiment 2 investigated the time spent reading ambiguous nouns that were either balanced, dominant / subordinate with relatively high frequency of use, dominant / subordinate with low frequency in a text that either had an informative topic heading or did not. For instance, readers would read an ambiguous text such as

> Every Saturday, four friends get together. When Jerry, Mike and Pat arrived, Karen was in her living room writing some *notes*. ... (p. 1021)

---

[3] Laying a foundation is but one part of the structure building framework, but the other parts are not as relevant here, and are omitted.

In the experimental condition informative headings such as “A Group of Friends Plays Gin Rummy” or “Rehearsal Section of a Musical Ensemble” preceded the text. These headings then primed specific senses of the ambiguous words. With a music heading, notes would likely be interpreted as notes for music.

In experiment 2 it was found that the facilitating effect of title context on reading times extended to both frequent and infrequent dominant senses, frequent and infrequent balanced senses, but only to frequent subordinate senses. That is, no less time was spent reading words that were disambiguated towards an infrequent subordinate sense when informative topic headings preceded the text compared to when there was no preceding topic heading. Evidently, the subordinate bias effect was overcome by title context in case of the frequent subordinate senses, but not for infrequent subordinate senses. That is, the effect of context appears to have been sufficient to cause sense selection in case of the more frequent subordinate senses. Experiment 3 verified that these results subsisted when control words were matched in frequency and length to the frequency of the subordinate sense and the length of the corresponding ambiguous word.

The results of Pickering and Frisson (2001) indicate that global context such as topic headings are useful in early incremental resolution of word senses. The senses used in Pickering and Frisson were mainly coarse grained (Senses in experiment 2 were from Twilley, Dixon, Taylor, & Clark, 1994). The topic headings and knowledge activations are conceptually similar, and so

It is however worth noting the results obtained by Wiley and Rayner (2000) might not apply fully to homonymic verbs. As noted by Pickering and Frisson (2001), the context effects they obtained on homonymous verbs were not evident until the next word, while previous research had found earlier effects of context on nouns. Pickering and Frisson suggest that the difference is due to a difference in the time course of lexical resolution of verbs and nouns. Note however that the contextual manipulation in Wiley and Rayner (2000) likely was quite strong. When manipulating the topic headings, they also caused many words and consequently sentences to have a topic-consistent meaning. Moreover, upon reaching ambiguous words at the end of the text, readers would have recently resolved quite a few ambiguous words towards a topic meaning, and so lexical disambiguation in a topic-consitent manner should be facilitated. The contextual manipulation of Wiley and Rayner is not at all lacking in ecological validity[4], but is perhaps similar to a different class of language comprehension situation than the experimental materials of Pickering and Frisson (2001). I.e. coherent texts about a topic with ambiguous words with a topic heading compared to more isolated sentences.

6.4.3 What senses are represented lexically?

The research discussed in the preceding section concerns lexical disambiguation. In word sense disambiguation research however, different terms are used. *Polysemy* denotes the case where

---

[4] Ecological validity is the degree to which an experimental setup is similar to the actual situations one seeks to investigate.

a word may have several related meanings, whereas *homonyms* are words that have several independent meanings associated with them[5]. Additionally, homonym level distinctions are seen as coarse grained distinctions, while polysemy to denotes finer grained meaning distinctions. That is, what sense distinctions are reflected in lexical representations? Moreover, how early are different types of sense distinctions made?

Klepousniotou (2002) investigated whether or not polysemic words are represented by distinct lexical items. An established finding in lexical ambiguity resolution is that contexts that activate the less frequent senses of a word increases the time it takes to recognize that word (see section 6.4.1). This research has primarily concerned homonym level sense distinctions, and establishes that homonym level senses are encoded on different lexemes. If polysemic senses are represented by different lexical items, similar competitive effects should result. However, if polysemic senses of a word are encoded on the same lexeme priming of both senses should reduce response times.

Klepousniotou (2002) found that there was a greater effect of contextual primes (of both primary and secondary sense) on word recognition times for metonymies (e.g. primes of mass and exemplar senses of "potato" (e.g. "some potato" and "a potato")) than for homonyms (e.g. primes of river and financial institution senses of "bank"). Interestingly, for a different type of polysemy, metaphorical extension, there was no difference between the effect of primes of different metaphorical senses (e.g. tail of airplane, tail of dog) and primes of homonymic senses nor of metonymic senses on word recognition times. According to Klepousniotou, these results occur because metaphorical exensions occupy a middle ground between full and entirely lacking lexicalization.

Following the findings I discussed in the last sections, lexical identities are resolved rather quickly, either by selective activation or selection processess. But how early are polysemous word senses resolved by context? Answering this question should have implications for the role of knowledge activations in resolving senses.

Pickering and Frisson (2001, Experiment 1-2) investigated the time course of contextual resolution of homonymic verbs (e.g. bore a hole and bore an audience (p. 570)) compared to polysemous verbs (e.g. launching a product or launching a rocket (p. 571)) using an eyetracking paradigm. Their results complement those of Klepousniotou (2002).

In experiment 1, partcipants were presented sentences containing homonymous verbs, where:

- The antecedent part of the sentence either primed the dominant sense of the verb or was neutral with respect to word sense, and the subsequent part of the sentence resolved the meaning of the verb to the dominant sense ((4-a) and (4-c)).
- The antecedent part of the sentence either primed the subordinate sense of the verb or was neutral with respect to werd sense, and the subsequent part of the sentence resolved the meaning of the verb to the subordinate sense ((4-b) and (4-d)).

---

[5] A different definition based on etymology exists, but is not adopted here, as it seldom used in psychology.

(4) a. Because he waffled all the time, Sam *bored* a couple of guests enormously at the reception last Thursday.
b. In order to hang up the speakers, we *bored* a couple of holes in the wall of the living room last night.
c. It was indeed Fiona who said that Al *bored* a couple of guests enormously at the reception last Thursday.
d. It was indeed Fiona who said that Al *bored* a couple of holes in the wall of the living room last night.

This manipulation allowed Pickering and Frisson to establish that there was an effect of the contextual manipulation, and that this effect was evident after having read the verb. In the example above, readers spent less time reading "a couple of" and "a couple of" when context primed the relevant meaning of "bored". (p. 562-564)[6]. The authors note however that effects of context on ambiguous nouns are apparent when reading the words themselves, indicating that lexical disambiguation is delayed for verbs compared to nouns (p. 563).

(5) a. In order to spy on the enemy, Russia *launched* a couple of satellites into the sky.
b. In order to sell more, the companies *launched* a couple of goods on the food market.
c. Over six days have passed since they *launched* a couple of satellites into the sky.
d. Over six days have passed since they *launched* a couple of goods on the food market.

When the experiment was redone (experiment 2) with similar materials using polysemous verbs however, the preceding disambiguating context had effects on reading times at the end of the sentence ("into the" and "on the" respectively).

The authors take these findings as evidence that although coarse grained sense distinctions concerning verbs are resolved later than coarse grained sense distinctions concerning nouns, they are resolved quickly in comparison with finer grained sense distinctions.

Although preceding context does appear to contribute to the resolution of fine grained sense distinctions, fine grained senses appear to remain underspecified for some time. It is likely that local sentence context is more important in resolving fine grained senses than coarse grained senses. Thus, knowledge activations have a lesser role in resolving fine grained senses than coarse grained senses, as knowledge activations represent global aspects of context.

From NLP research there is evidence to suggest that activated knowledge structures only discriminate homonym level senses. Poor interrater reliability obtains when characterizing natural language using a fine grained-sense inventory (Navigili, 2009, doublecheck). This reliability is improved with coarse grained sense distinctions.

In an experiment using human judgements of word senses, Gale, Church, and Yarowsky (1992) obtained evidence that multiple occurences of a word in a discourse (defined as a co-

---

[6] First pass reading times.

herent piece of text) tend to have the same homonymic sense[7]. This notion is denoted the *one sense per. discourse heuristic*. Participants were found to judge pairs of key words in context (KWIC) as having the same sense when they came from the same discourse, and as having different senses for KWIC-pairs with different senses from different discourses. The level of discrimination at which senses are indistinguishable in a single discourse appear to align well with the human capacity to distinguish word senses.

In a follow up study, Krovetz (1998) found evidence that the one sense per discourse heuristic does not extend to polysemous meanings. Krovetz investigated intances of multiple senses per discourse in a corpus tagged with WordNet (known for including fine grained sense distinctions), and found that these sense distinctions were overwhelmingly polysemous.

In summary, coarse grained sense distinctions are likely encoded as lexical distinctions, while fine grained sense distinctions are not. Lexical identities are resolved relatively quickly after a word is presented, but finer grained sense distinctions are not computed until a later stage of processing. Early resolution of word senses induced by states of knowledge activation is likely constrained to broad sense distinctions.

6.4.4 Use of domain vectors in incremental processing

Magnini, Strapparava, Pezzulo, and Gliozzo (2002) posited that information about the semantic domain of a text is useful in disambiguating the senses of words in that text. Texts were assigned feature vectors describing the strength of their relationship to various domains, such as politics or sports. In training the system, senses were associated with the sum of the domain vectors of texts where they occur, or if too few examples were available, the domains of the sense induced from WordNet Domains weighted by the frequency of these domains in a corpus. The senses of words in text were predicted by comparing the domain vector of the text containing the word to the domain vector of the possible senses of the word. Using these features, the senses of words were predicted at a level that was comparable to the highest ranking systems in SENSEVAL-2, supporting Magnini et al. (2002)'s hypothesis. Boyd-Graber, Blei, and Zhu (2007) obtained similar results in an unsupervised paradigm. Domain vectors appear to be promising candidates for elements of A.

Equipping the states of the model with domain vectors or other representations describing the topic of a text should likely approximate representations global context as represented in the minds of human beings. More importantly, the model will be able to use such representations in the same way humans do, in resolving coarse grained word senses at an early stage of processing. Moreover, activations of knowledge, possibly instantiated by domain vectors, are likely useful in this regard.

The state of semantic activation can be considered as a kind of preprocessing mechanism.

---

[7] The terminology in Gale et al. (1992) could be slightly confusing, as they use the term polysemy to denote what is here denoted homonymy.

Intuitively, great amounts of linguistic ambiguity are resolved before comprehenders even begin to read or hear a many sentences by virtue of the state of activation of knowledge a comprehender is in. This might explain the fact that humans are oblivious to ambiguities that are encountered by NLP-programs. These may not reach activation, either because context favors an alternative meaning, or the meaning is subordinate and does not reach activation unless context lifts it. There is an obvious computational advantage in the form of a reduced branching factor in later processing by the addition of such a preprocessing stage. By breaking the assumption of the standard pipeline that context should be integrated only in the very last stages, more effective programs may be achieved. An NLP application could for instance use metadata or indeed a first pass with a text classification procedure in order to set the correct state of semantic activation when processing the text with the main procedure.

The computational explosion resulting from a large sense inventory with many low frequency (coarse grained) senses of words ought to be mitigated by the addition of a state of knowledge activation. When new senses are added, one should make note of the knowledge domains where they occur, ensuring that they are readily available when they could be used. According to the reordered access model then, these senses should be considered very late in contexts that do not activate them. In such contexts, many rounds of reanalysis might be necessary in order to access such a sense.

Fine grained sense distinctions however, should be underspecified in initial meaning representations; their resolution deferred until later stages of processing.

6.4.5 Consequences for the model

In the preliminary model, alternative continuations of a sentence were labelled by different syntactic decisions. In the model extended with meaning representations, further alternatives were introduced, motivated by differing semantic decisions on word senses. In a given meaning representation, parser state, one must decide how to incorporate a new word both syntactically and semantically.

The findings on the time course of fine grained senses however, indicate that increments should only be indexed by coarse grained word senses, as humans appear only to commit to coarse grained distinctions as they build meaning representations incrementally. Furthermore, meaning representations must be compatible with figurative as well as literal interpretations, as there is no evidence that comprehenders must revisit incremental decisions in order to obtain a figurative interpretation.

The evidence strongly suggests that states of knowledge activation representing topic or domain level context are highly useful in resolving coarse word sense distinctions at a very early stage. In the model, different states of knowledge activations will lead at least to differently ranked semantic alternatives, resulting in different alternatives chosen in incremental processing, and in different alternatives being explored when sentences are reprocessed. The introduction of a state of knowledge activation to the model means that decisions that in the standard pipeline would have to be postponed or explored in parallel can be made immediately. Even when the initial decision turns out to be problematic, the introduction of knowledge activations means likely alternatives will be considered earlier.

If two stage models hold, the rankings on syntactic aspects of increments should be unchanged when knowledge activations are introduced. After all, only syntactic properties of words should change how they are incorporated into a developing syntactic structure, not how the given word connects to knowledge structures. However, there is preliminary evidence that the parser does take into account subcategorization preferences of verbs, and that such distinctions are sometimes coded as relatively coarse grained sense distinctions.

The next section discusses preliminary findings that indicate that global context (mediated by states of knowledge activation in the model) influence syntactic decisions through selective activation of verbs with different subcategorization preferences.

6.5 Use of states of knowledge activation in the model

Alternatively, with weaker contexts and more subordinate senses, A may lift the sense such that it is consider earlier than otherwise in reprocessing. Such effects do not appear to have been researched much, and are not discussed further here. The paradigm of Cai et al. (2012) appears to be a promising candidate for such a study.

A third option is that a subordinate sense is lifted to a level where it can compete with a dominant sense. The question of whether such competition translates into corresponding syntactic competition is highly contentious. On face value, competetion between lexical representations may yield disambiguation results at too late a stage in the time course of incremental processing to be reflected in syntactic increments, at least if competition effects cause processing difficulty.

6.5.1 Selective access to verbs with different subcategorization profiles

One way in which MacDonald et al. (1994) proposed that context may influence syntactic parsing is through selective activation of senses pertaining to a context.

Some verbs have different senses that have differing subcategorization profiles; if the concrete interpretation of “found” is activated, perhaps in a story about pirates digging for treasure, we would expect a sentence fragment “They found” to continue (at some point) with a direct object. If a commision is looking into a matter, we would expect a complement.

Hare, McRae, and Elman (2003) selected 20 verbs likely to be subject to such an effect, and found corpus evidence that senses promoting complement or direct object subcategorization did indeed predict the corresponding syntactic structure. Moreover, participants exposed to a contextual prime for either meaning tended to complete sentence fragments with either direct objects or complements depending on activated sense. These results, although suggestive, are not sufficient to establish an effect of such contextual primes on online syntactic decisions.

Using an eyetracking paradigm, Hare et al. (2003, p. 290) primed participants with pairs of sentences such as:

(6) a. The intro psychology students hated having to read the assigned text because it was so boring.
    b. They found (that) the book was written poorly and difficult to understand.

(7) a. Allison and her friends had been searching for John Grisham's new novel for a week, but yesterday they were finally successful.
    b. They found (that) the book was written poorly and were annoyed that they had spent so much time trying to get it (p. 291).

The probe sentences (6-b), (7-b) had the complementizer "that" in half of the cases. Without "that", the probe sentences are locally ambiguous towards a direct object reading of "the book", which is finally disambiguated as a complement by the word "was". The difference between reading times over different parts of the probe senteces was compared in the two conditions. The difference between the reading times between disambiguated (complement) and locally ambiguous conditions was greater for (7), than for (6), indicating that the comprehenders had been led to choose the direct object reading of "the book" in (7).

Unfortunately, little follow up research has been done, and the findings of Hare et al. (2003) do not appear to have been investigated further.

Evidence that syntactic decisions are informed by subcategorization properties of verbs provide indirect support for Hare et al. (2003).

Experiment 3 in Kizach et al. (2013) investigated whether or not initial syntactic decisions are influenced by subcategorization properties of verbs. They found that verbs that subcategorize for complement phrases, but not determiner phrases (i.e. direct objects), also cause readers to be more likely to adopt complement readings of subsequent words prior to disambiguating information. When verbs subcategorize for both complement phrases and direct object readings, they are less likely to adopt complement readings. In the sample materials below (in Danish), "hørte" (Eng. "heard") subcategorizes both for complement phrases and determiner phrases, whereas "tænkte" (Eng. "thought") subcategorizes for complement phrases only.

(8) a. Filosoffen *hørte* forelæsningen om etik vakte begejstring.
    b. Filosoffen *tænkte* forelæsningen om etik vakte begejstring.
    c. Philosopher-the heard/thought lecture-the about ethics evoked enthusiasm. (p. 7, emphasis added).

Participants were presented the words one by one in a paradigm known as a maze-task (Forster, Guerrera, & Elliot, 2009). In the maze tasks, words from the sentence are presented one by one, paired with distractor words that cannot be incorporated into the syntactic structure, and participants indicate with buttons, for each pair of words, which word is the word that continues the sentence.

Participants that read (8-b) took longer selecting the correct word immediately following the manipulated verb than participants that read (8-a). Participants that read (8-a) took longer selecting the correct word when reading the the verb disambiguating the direct object reading (i.e. "vakte" ("evoked")) than participants that read (8-b) (p. 8). According to Kizach et al., the difference in response times for the word immediately following the critical verb are due to the fact that complement readings are harder to construct. However, as the authors point out, this delay could also be interpreted as brief reanalysis resulting from the direct object reading of "forelæsningen" being initially chosen in (8-b), but then disregarded and reanalyzed in the direction of a complement phrase. The increased response time at the disambiguating verb "vakte" for *both* is understood by the authors as a classic garden path effect. Parcicipants were more likely to choose a direct object reading in this condition than in *one*. If the increased reading time of "forelæsningen" indeed is due to increased complexity and not reanalysis after an initial direct object reading, the results indicate that the parser uses subcategorization information to make syntactic decisions.

Kizach et al. (2013) note that the findings of Staub (2007) support the interpretation that the increased response time does not result from an initial direct object reading with subsequent reanalysis. Experiment 2 in Staub (2007) investigated exactly these conflicting explanations. They compared reading times for words regions disambiguating a clause towards a complement reading when both the syntactic analysis of the sentence thus far and the subcategorization of the main verb ruled out a direct object reading, and when only the subcategorization properties of the main verb ruled out the direct object reading.

(9) a. When the dog arrived the vet and his assistant went home.
b. When the dog arrived, the vet and his assistant went home.

(10) a. When the dog arrived at the clinic the vet and his assistant went home.
b. When the dog arrived at the clinic, the vet and his assistant went home.

In (9), the subcategorization of "arrived" dictates that no direct object reading exists for "the vet"'. In (10), such a reading of "the vet" is not only incompatible with the subcategorization of "arrived", but also with any reading of "When the dog arrived at the clinic". If subcategorization information does not cause the parser to decide to initially attempt to incorporate "the vet" as a complement, but does so only with an added process of reanalysis, but incorporates the clause as a complement as the initial analysis in (10), readers should take longer reading "the vet" in the subcategorization only condition. Moreover, the difference between reading times of "the vet" with or without a preceding disambiguating comma should be greater in the subcategorization only region. However, Staub (2007) found that there were no such effects (p. 557). Hence, it is likely that the effects of subcategorization properties of a verb on syntactic parsing are not due to reanalysis, but rather due to an effect on initial decisions.

Sense-dependent subcategorization represent a very indirect route by which semantic information can influence syntactic processing. In this pathway from semantic information to syntactic information, semantic information (word sense distinctions) appears to be translated into syntactic information (subcategorization) before it influences syntactic decisions. Thus, such a semantic influence on syntactic is not incompatible with a view of syntactic processing as distinct from semantic processing. One need not propose a constraint based view of language processing without a modularized parser to incorporate such a pathway.

The term *syntactic ambiguity* in psycholinguistics appears to refer to syntactic ambiguities that may be percieved rather easily by human comprehenders. The syntactic ambiguities discussed in NLP however, are often ambiguities that comprehenders need special training to notice. It is important then, to proceed with some caution when considering psyhcolinguistic studies on how syntactic decisions are made. Few, if any, studies of syntactic processing have considered how comprehenders manage to exclude those alternative syntactic continuations that are not easily perceived.

One possible explanation of why many syntactic ambiguities are not registered by human comprehenders is that the word sense ambiguities that produce some such ambiguities are simply not available for consideration; their underlying memory structures are not sufficiently activated by context, and they are too infrequent to be considered on their own merit, or context produces selective activation of alternative senses.

The subordinate bias effect discussed in section 6.4.1 gives further insight into how states of knowledge activation can help syntactic processing. According to the research on the subordinate bias effect, many, infrequent senses are not considered in ordinary contexts. In special contexts however, such senses may be lifted to the level where they are considered alongside more frequent senses.

In terms of the model, this means that extending the carrier with states of knowledge activation (i.e. the set A) can change the order in which alternatives are ranked by the underlying state space (denoted $incr3\_alt$). With a context, a sense may be lifted to the level where it is selectively activated, or is considered earlier in reprocessing.

Chapter 7

Extending the model with context models

The present chapter is about the early role of context in language comprehension, and how the model can be extended to correspond to these findings. With the term *context model* I refer to a model of the situations and events being described in preceding text, and not models of the communicative situation or other variables external to the message being conveyed. That is, the situation models discussed in 6.1.1, as they refer to text-level messages. For instance, when reading a novel, the context model should correspond to the participants, situations, and events described by preceding text.

I will argue that a context models should be made part of the states of the model. Adding such information is important if the model is to process language in a psycholinguistically plausible way. It appears particularly important in making knowledge based constraints less rigid.

There are several types of evidence that indicating that comprehenders keep inner representations of the situation described thus far in a text, and that such representations are drawn upon during on-line language comprehension. First there is evidence that the N400 effect is sensitive to fit of the words in a sentence with structured representations of preceding context. This evidence is discussed in section 7.1.

Second, comprehenders have been found to use visual scenes to guide language comprehension 7.2. This guidance is mediated by inner representations. Apparently, this guidance develops with time, as language comprehension in young children does not appear to be similarly guided by visual scenes.

I argue that the context models should be added to the carrier (i.e. the states) of the model. It appears that comprehenders resolve referring expressions in text to entities in such a model in on-line processing. Correspondingly, implementations of the model should attempt to do a task similar to the NLP task of coreference resolution with respect to the context model as language is incrementally processed. This contrasts to the standard pipeline, where context-dependent tasks such as coreference resolution are postponed until after both syntactic and semantic processing are completed.

If referring expressions were resolved against context models, computer programs implementing the model should be better equipped in handling figurative expressions that break selectional restrictions based on literal, lexically derived word meanings. As such, the addition of context models to the states of the model can override situations where general knowledge wants to throw an error. Moreover, the concept of semantic implausibility can be extended to cover not only implausibility in terms of general knowledge, but implausibility in terms of a context model. As such, new situations where the mechanism throws an error are introduced. Additionally, incremental coreference resolution should interact with syntactic processing, improving the quality of syntactic decisions.

7.1 The integration view of the N400 effect

According to Lau et al. (2008, p. 921) “One longstanding view of the N400 effect is that it reflects the process of semantic integration of the critical word with the working context”. Moreover, such working contexts are thought to be both sentence local and from the wider discourse (Hagoort & van Berkum, 2007, p. 803).

van Berkum, Hagoort, and Brown (1999, Experiment 1) manipulated the fit of words in written sentences with previously established context. An N400 effect obtained, with a higher amplitude for critical words in the context-incongruent condition. When sentences were presented without the context, the difference between the N400 amplitudes for the incongruent vs. control words was significantly smaller (Experiment 2).

Hagoort and van Berkum (2007) claim that such findings contradict a two step model of language processing where context is brought to bear on sentence meaning only after its local meaning has been constructed (p. 801).

Previous discourse may establish assignments of properties to entities that they do not ordinarily have. For instance, a rock might be treated as an agent in a discourse. Nieuwland and van Berkum (2006) found evidence that such assignments may influence language comprenhension on-line incremental language comprehension. In their second experiment, Nieuwland and van Berkum (2006, p. 1104) presented participants with sentences that violated lexical knowledge and sentences that were consistent with lexical knowledge. For instance, the sentence “The peanut was salted” is not consistent with lexical knowledge, whilst the sentence “The peanut was in love” is. No preceding context was presented. In line with previous N400 research, the sentences with violations of lexical knowledge elicited greater N400 responses at the critical words (e.g. “salted” and “love” in the previous example). However, when participants were first presented a story that licensed the lexical violation (a story assigning the peanut agency), the results were reversed. N400 responses were in fact greater for sentences that did not violate lexical knowledge than for those that did. The authors concluded that the canonical, lexical meanings of words can be overruled at an early stage of processing by meanings arising from structured representations of context.

The results also speak to the comprehensiveness of the lexical view of the N400 effect. The results in the above experiment require an inner representation of context to be structured, and so rules out an unstructured activation of lexical knowledge as a comprehensive explanation.

If the integration view of the N400 effect holds, it should be responsive not just to activation of knowledge structures, but the role these knowledge structures play in preceding context. Otten and van Berkum (2007) investigated this prediction. They presented two different types of of experimental materials such as the following examples:

(1) a. The manager thought that the board of directors should assemble to discuss the issue. He planned a *meeting/session* where the staff members involved would be present as well.
    b. The manager thought that the board of directors need not assemble to discuss the issue. He planned a *meeting/session* where the staff members involved would be present as well. (p. 168, emphasis and "/" added for brevity).

It was verified that the word "meeting" is generally the term which is expected after participants are primed with the scenario suggested by the first sentence, while "session" is unexpected, as verified by Otten and van Berkum (2007) with cloze probabilities. In the first pair (1-a), the expected target word "meeting" is consistent with the general scenario presented in the preceding sentence, and indeed with the message in the preceding discourse. In the second sentence (1-b) however, "meeting" is not consistent with the message in the preceding discourse, but is consistent with the scenario.

If the N400 is only sensitive to manipulations of what kind of topic or scenario is suggested by preceding context (as suggested by the lexical view), there should be no difference in the N400 effect of manipulations of "meeting / session" when message fit is also manipulated. Otten and van Berkum (2007) did not find statistically significant differences in the N400 effects (p. 169). There were however differences in the neural origins of the N400 effects in the two pairs of sentences. The result suggests that different neural systems contribute to the integration-related and the lexical N400-effects respectively[1].

Although there is evidence for effects of context models on incremental processing, it is still unclear how complex such effects are. The short interval of the 200-400 ms window likely constrains the complexity of the computations. A precise, empirically sound characterization of how context models influence ongoing processing is not yet available.

7.2 Effects of visual context on incremental processing

The present section provides evidence that suggests that comprehenders maintain a model of visual context, and that this model is brought to bear on incremental processing. Specifically, syntactic decisions during the comprehension of utterances about perceptually available situations

---

[1] Lau et al. (2008) discusses the possibility that there are different substrates of the two N400 effects discussed here.

can be guided by visual context. The effect appears to result from the fact that comprehenders resolve referring expressions to entities in contexts.

Researching comprehension of utterances about perceptual situations is somewhat easier than researching comprehension about situations that are not immediately available to the senses, as online interaction with perceptual displays can be monitored by tracking eye movements during language comprehension. Such eye movements can reveal incrementally chosen syntactic analyses.

Comprehending utterances about perceptual data is likely a model situation for language comprehension in general, as connecting language with perception is how we first use language. Shared attention to objects (coupled with verbal description) between a caregiver and an infant is thought to be an important process in language development (Tomasello, 1988). The *visual world paradigm* is a researh paradigm that can reveal the effects of context on language processing.

Eberhard, Spivey-Knowlton, Sedivy, and Tanenhaus (1995) reviewed eyetracking research indicating that language comprehenders incrementally restrict the domain of possible referents during language comprehension, and that domains of possible referents can guide parsing decisions. Participants were told to pick up an object, the name of which was some times similar to one of the other objects in a display. Eye tracking data revealed that visual attention was guided towards the referent before the entire name of the object had been presented when there was no competition, but after the entire name had been presented when there was an object with a similar name (Experiment 3). A representation of the available objects is presumably maintained (similar to the context model). Moreover, comprehenders attempt to resolve referring expressions to such objects as soon as possible.

Experiment 5 discussed in Eberhard et al., 1995 provided evidence that the domain of possible referents in a display can guide syntactic analysis. Different attachments of a prepositional phrase in the ambiguous sentence “Put the saltshaker on the envelope in the bowl.” (p. 428) give rise to interpretations where the the saltshaker should be put on an envelope (in a bowl), or the saltshaker is currently on the evelope and should be put in the bowl. The unambiguous sentence “Put the saltshaker that’s on the envelope in the bowl” do not give rise to such interpretations. Participants were presented displays containing either:

1. Only one saltshaker which was positioned on an envelope. Another envelope, without a saltshaker on it. A bowl.
2. Two saltshakers, only one of which was positioned on an envelope. Another envelope without a saltshaker on it. A bowl.

When participants were presented display 1 and the ambiguous sentence, they tended to look at the envelope (without a saltshaker on it) after hearing“on the envelope”, apparently interpreting the envelope as the goal of the movement. This did not happen if they heard the unambiguous sentence. When participants were presented display 2 there was no effect of sentence ambiguity on looking at the envelope. In short, the evidence appears to suggest that incremental parsing decisions are being determined by the perceptual display (p. 429). The findings have since been replicated (cf. Huettig, Rommers, & Meyer, 2011, p. 155). However, the behavioral measures used in the studies (looking at objects) do not measure syntactic commitments directly, but rather, interpretations arising from syntactic commitments. Possibly, the effects on looking are due to direct semantic processing. More direct measures of syntactic commitments, such as a garden path effect would establish the effects on parsing more directly. Such measures are however hard to obtain in the visual world paradigm.

Chambers, Tanenhaus, Eberhard, Filip, and Carlson (2002) found evidence that the domain of possible referents of expressions of natural language is updated continously during sentence processing, and not just at sentence or clause boundaries. Using an eyetracking paradigm, participants were presented with displays containing objects and one or three containers. They found that instructions to pick up a given object and place it *inside* another object (e.g. "the can") led participants to pay more attention to a single container when only one container was present, but not when three were present. Instructions to put the object *below* an object did not yield a difference in attention to a single container when only one was present. (Experiment 1). Critically, the eyetracking data revealed that the container-noun could not have influenced attention in the given time frame, indicating that the pool of possible perceptual referents was updated before the sentence was fully presented.

Altmann and Kamide (2009) provides evidence indicating that the evidence presented above are mediated by internal representations. In experiment 2, participants were presented clip-art pictures (on a monitor) of a scene with a person and some objects. In the example below, there was a picture of "man at desk, floor fan, drinks can, wastebin, swivel chair" (p. 69). Then, one of two possible sentences were presented (auditorily), either (2-a) or (3-a).

(2) a. The office worker will drag the dusbin right next to the fan.
b. Then, he will grab the can, and chuck it violently into the dustbin.

(3) a. The office worker has just dragged the dustbin away from the fan.
b. Now, he will grab the can, and chuck it violently into the dustbin. (p. 69)

In the picture, the dustbin was away from the fan, so only the situation described in (2-a) differed from the pictured situation. The picture was removed before the final sentence (2-b) / (3-b) was presented. Eye-movements were monitored. When either target of the movement ("the dustbin") was read, participants were more likely to look at the place on the monitor where the object had been moved, (i.e. next to the fan) if they were presented sentence (2-a) than if they were presented (3-a). In experiment 2, the findings were reproduced, but the picture was removed before any sentences were presented (p. 65). Evidently, the participants kept updated representations of the situation being described, and this information was drawn upon during incremental language comprehension.

7.3 Visual context in young children

Relatively recent research has investigated the performance of young children (5 years old) on the visual world paradigm employed by Eberhard et al. (1995). The findings give insight into how the effect of context can be learned. As has been discussed, strong cues from visual context can guide syntactic increments. However, as will be discussed shortly, the parsing decisions in young children do not appear to be guided by context in the same way.

Trueswell et al. (1999, Experiment 1) presented children with a visual scene similar to that of Eberhard et al. (1995) and tracked their eye-movements. In this experiment, there were pairs of stuffed toy animals, one of which was on a napkin, a napkin without an animal, and an empty box. The children were presented pairs of sentences (read aloud) such as:

(4) a. Put the frog on the napkin in the box.
b. Put the frog that's on the napkin in the box. (p. 100)

The first sentence is locally ambiguous after "on the napkin". Discourse context was manipulated towards an analysis of "on the napkin" as a modifier of "the frog" by either having two frogs as the animals, or having in addition to one frog, a different animal. Experiments with adults has shown such a manipulation reliably shift the incremental reading towards a modifier reading. However, there was no effect of the discourse context manipulation on incremental syntactic decisions. The children tended to interpret "on the napkin" as the destination of the frog. The children were much more likely to look at the napkin (the incorrect destination) in ambiguous condition than in the disambiguating condition. The children started looking at the incorrect destination as soon as "on the napkin was presented". Critically, this tendency was not significantly different as a response to the discourse conditions. Consistent with these findings, the children also tended to act in accordance, moving the frog to the napkin instead of the box.

Similar results were also found by Snedeker and Trueswell (2004). Weighall (2008) replicated the findings of Trueswell et al. (1999), but with stronger controls. It is worth noting that there is evidence that the initial parsing decisions of children are guided by verb biases (Snedeker & Trueswell, 2004). As such, interpretations of these results in the direction of exclusively syntactic principles guiding incremental analysis are not warranted. The findings give insight into how language comprehension develops. Evidently, lexical and syntactic constraints (encoded by P) have a stronger role in determining the P as language is learned. Only later does the component C appear to play a role in determining P. The results speak to how an implementation consistent with the model can be trained. What has not been investigated is to what degree contextual factors influence semantic increments in children. Abstracting away this consideration, a reasonable way of training a computer program based on the model appears to be to begin with a parser that is not sensitive to the influence of context, and then to distinguish domain and context dependent situations where the behavior of the parser diverges from the canonical case.

Additionally, Trueswell et al. (1999) found that “a child’s commitment to an interpretation seemed almost deterministic or ballistic, showing little or no ability to return to an earlier state” (p. 121-122). The children did not appear to reanalyze the sentence after their initial anlyses turned out to be erroneous, as they did not tend to revise their choice of one of the two frogs when they selected the wrong one (i.e. the frog that was not on the napkin). Possibly however, they might have reanalyzed the sentence, but not have been able to correct ongoing behavior resulting from the initial analysis. Weighall (2008) controlled for this possibility by delaying the acting out-part of the experiment. The children were still found to act in a way consistent with their initial interpretation of the sentence (p. 90). The initial iterative control structure proposed in section 4.2.4 has significant overlap with such an idea. Note however that instead of reporting errors, young children appear to proceed with ungrammatical analyses.

7.4 Implications for the model

The purpose of this section is to discuss the implications that the findings discussed in this chapter have for the model. In terms of NLP tasks, a basic finding from the previous sections is that coreference resolution happens during incremental processing, and not in a second stage after the sentence has been processed.

Coreference resolution is the task of relating those references (possibly including implicit references) in a text that refer to the same entity or event, and not relating those references that do not. Context models and developing representations are likely used to resolve coreferent mentions as they are encountered in text.

I will model coreference resolution as a commitment made by language comprehenders along with the semantic and syntactic commitments already discussed. Thus, a set of coreference commitments $Coref$ are made part of the model. The new functor is $G4$, given below:

$$X \mapsto (E + (Coref \times Sem \times Syn \times X))^W$$

The carrier is updated to keep track of context models, represented by a set C. The new carrier is then $C \times A \times M \times P$, with a new coalgebra $(C \times A \times M \times P, incr4)$. In order for coreference / context model commitments to truly be incremental, $incr4$ should be subject to monotonicity requirements on C, as well as the existing requirements on M and P.

It is still an open research question in psycholinguistics whether or not coreference commitments require reprocessing to be overturned. It is likely that the process of revising coreference commitments differs from syntactic reprocessing. As there is very little evidence to go on, I will not consider this question any further.

7.4.1 Changes in $incr4$

There are three main ways in which $incr3$ and $incr4$ differ. First, as we have seen in the preceding section, syntactic decisions may change with the addition of a context model.

Second, depending on the context model, some states of the coalgebra that incremented successfully may now lead to errors. This is similar to what happened when meaning representations M were added. Now however, instead of introducing errors that somehow result from implausible or impossible meanings in terms of world knowledge, one can introduce errors that result from incompatibility of meaning with the particulars of a context model.

Somewhat surprisingly, the addition of a context model, and incremental coreference resolution may lead to the removal of semantic errors introduced by the addition of M. Evidence to this effect was discussed briefly in 7.1. That is, the introduction of a context may override general knowledge. Further evidence on such an effect is discussed in the following section.

7.4.2 Dynamic Selectional Restrictions

A *selectional restriction* of a predicate is a requirement that arguments be well-typed, i.e. correspond to typing requirements of the argument slots they occupy (Jurafsky & Martin, 2009, p. 661). The term *thematic role* is often used interchangably with an argument slot of a predicate with a corresponding semantic restriction. Selectional restrictions are indexed by / encoded as properties of predicates, meaning that every occurence of a predicate is required to be well typed. Selectional restrictions have traditionally been used to automatically disambiguate readings of sentences. They are one theoretical way in which knowledge could influence language comprehension. There are however some problems with selectional restrictions:

(5) a. It is impossible to converse with a rock.
  b. In his manic periods, he would carry out full conversations with large rocks.
  c. But the rock cried, I can't hide you.[2]

The sentences above can be understood with ease by language comprehenders, and yet they appear to violate selectional restrictions. The requirement that arguments be well-typed appears too strong. Examples (5-a) and (5-b) are inspired by (Jurafsky & Martin, 2009, p. 682), and reflect problems dealing with some negated sentences and problems covering the entire range of use of predicates. Example (5-c), reflects problems with coverage of metaphor. In this section, we will look closer at breakages of selectional restrictions due to metaphor.

In order to ameliorate such problems, the concept of selectional *preferences* was proposed by Resnik (1993). In short, predicates now do not require arguments to be of certain types, but have weighted preferences for arguments in different slots. These preferences are determined by how much change in uncertainty (i.e. information) about the types an argument is associated

---

[2] From the traditional song Sinner man (Simone, 1965).

with learning the identity of a predicate (p. 54).

The context model might enable stronger associations between predicates and arguments than selectional preferences. Note the example story from Nieuwland and van Berkum (2006, p. 1106)

(6) a. A woman saw a dancing peanut who had a big smile on his face. The peanut was singing about a girl he had just met. And judging from the song, the peanut was totally crazy about her. The woman thought it was really cute to see the peanut singing and dancing like that.
b. The peanut was *salted/in love*, and by the sound of it, this was definitely mutual. He was seeing a little almond.

The property of being in love presumes sentience on the part of the peanut. Being salted is something done to inanimate objects. As was discussed in section 7.1, readers did not expect the peanut to be inanimate (i.e. salted), but did expect it to be sentient. Indeed, it would appear that the discourse model of the participants overruled lexical knowledge about peanuts, granting it the property of sentience.

Upon encountering such an example in text, the distributional method of Resnik (1993) resorts to a weakening of the association between the first argument slot of `in_love`, and arguments typed by sentience. Instead, such a case can be handled by adding to the discourse model a typing of the peanut with sentience. In a sense, the typing of arguments is weakened by becoming dynamic (i.e. state dependent). Thus, representing discourse context is likely to mean that the associtation of predicates to argument types does not need to be weakened as much, and that such constraints on interpretation become more robust in terms of metaphorical language.

Such (strong) associations of predicates with the types of their arguments may be of use when inferring metaphor and figuration. When comprehenders encounter ill-typed arguments of predicates, an error should be thrown. Depending on context, for instance the availability of alternative syntactic analyses, one might induce an altered discourse state. Indeed, findings to this effect were made by Nieuwland and van Berkum (2006, Experiment 1). They compared stories identical in all respects except mentions of the protagonist, which was either animate or inanimate, in which case it violated selectional restrictions. For instance, participants were either presented stories about a sailor in therapy or a yacht in therapy. At the first mention of the protagonist, there was an N400 effect, with the mention of the inanimate protagonist (experimental condition) exhibiting a greater N400 amplitude than the animate protagonist (control). With successive mentions however, the N400 effect decreased, and was not present in the mention of the protagonist in the fifth sentence.

For instance, the in the beginnig of a story, sentences such as (7) would lead comprehenders to change their discourse model, assigning agency the yacht. As more consistent sentences are presented, they likely grow more confident in this judgement.

(7) Once upon a time, a psychotherapist was consulted in her home office by a yacht with emotional problems. (p. 1100)

I have not provided implementation details as to how context models are be represented, or precisely how and when such changes in the properties of discourse participants should be made. Such an implementation is not at all trivial. Knowledge representation is a major, unsolved problem of artificial intelligence. The architechture of language processing proposed here however, provide a natural slot for (co)reference resolution against context models, and their influence on incremental processing. Adding such a facility will help programs consistent with the model make selectional restrictions less brittle when processing stories and other text involving anthropomorphism or other violations of semantic knowledge.

Chapter 8

Summary

The main result of the project is a high level model for programs that understand natural language. I summarize this model below, and contrast it with the standard pipeline. The first, major difference between the standard pipeline and the proposed model is that on the sentence level, the proposed model consists of multiple iterations of syntactic, semantic and context-based processing. This means that different types of information are not considered before others *on the sentence level* in the proposed model. The standard pipeline on the other hand, does all syntactic processing before semantic processing, and all semantic processing before processing in terms of context. This fact means that the proposed model can do reprocessing (backtracking) based on the meaning of the sentence before it has completed a sentence level analysis. The standard pipeline however, completes sentence level syntactic analysis before any backtracking prompted by meaning can occur. I have proposed a naïve reanalysis algorithm for the proposed model. This algorithm consists of a basic process that attempts to incrementally process the sentence, producing for each decision a set of alternative decisions, weighted by probabilities. If processing fails either due to syntactic or semantic problems, reprocessing kicks in and restarts the basic process from the best of the alternative decisions.

As a *word level* pipeline, I assumed, based on psycholinguistic findings, that syntactic increments have dominance over meaning increments. I encode this dominance as a simulation between the parser with meaning representations $incr2$ defined in chapter 5, where semantic labels are forgotten, and the purely syntactic coalgebra $incr$ defined in chapter 4. Moreover, as proposed in chapters 6 and 7, context on the global level (topic) and a structured representation of context have early influences in the proposed model. I claim that this early influence of context can resolve coarse grained meaning distinctions quite early and that it can guide parsing. As such, it reduces the problem of computational complexity resulting from a buildup of syntactic and semantic alternatives associated with the standard pipeline. Moreover, an early influence of context can make semantic errors more dynamic, in that it allows a context to override situations which in general lead to a semantic error, and to cause a context-related error where there generally speaking is no semantic error. This means that programs that process language in a way that is consistent with the model are better able to handle figurative phenomena involving violations of general knowledge and general semantic constraints on interpretation.

8.1 Further research

Psycholinguistics is a very broad field. In the present text, I have attempted to summarize important findings on a fairly broad assortment of research topics. There are however entire subfields of inquiry in psycholinguistics that may well be relevant to natural language processing that are not discussed in the present text. Further research should adjust and add components to the model according to such research. Discourse information, such as information about about speaker intentions are not taken into consideration at present. Modelling the early influence of such information may be important in covering figurative language more adequately. Most of the psycholinguistic research that has been reported on was conducted in English or Dutch. This bias may limit the generality of the findings, a property that is in turn inherited by the proposed model.

The model is largely silent on what happens in integration stages, and instead focuses on what happens in early parts of processing. More work is needed on modelling late stage processing, how it differs from the standard pipeline, and how it can be of use in NLP.

The present text represents a first attempt at coalgebraic modelling of the architechture of human language processing. I have attempted to apply a mathematical formalism ordinarily used in computer science to psycholinguistics, with applications in natural language processing in mind. A weakness of the present project is that the model developed has not been implemented and tested. Experiments with actual implementations have to be conducted in order to assess the value of the model for NLP. Moreover, the computational complexity of the proposed model was not analyzed and compared to the standard pipeline in any precise way. Further research should explore these properties, again in order to assess the value of the model for NLP.

There is also another possible direction for further research. The concept of a coalgebra could be used as a language for summarizing and comparing the architechture of programs that process natural language, irrespective of psycholinguistic plausibility. It could allow more systematic and fine grained comparisons of approaches to problems in natural language processing, elucidating patterns in performance arising from structural properties of programs. At present, this appears to be missing in NLP.

Bibliography

Allen, J. (1995). *Natural language understanding* (2nd ed.). Redwood City, CA, USA: Benjamin-Cummings Publishing Co., Inc.

Altmann, G. T. & Kamide, Y. (2009). Discourse-mediation of the mapping between language and the visual world: Eye movements and mental representation. *Cognition*, *111*, 55–71. doi:10.1016/j.cognition.2008.12.005

Barr, M. & Wells, C. (1999). *Category theory for computing science* (3rd ed.). Montréal, Canada: Les Publications CRM.

Bates, M. (1995). Models of natural language understanding. *Proceedings of the National Academy of Sciences of the United States of America*, *92*, 9977–9982. doi:10.1073/pnas.92.22.9977

Bever, T. (1970). The cognitive basis for linguistic structures. In J. Hayes (Ed.), *Cognition and the development of language* (pp. 279–362). New York, NY, USA: Wiley.

Binder, J. R., Desai, R. H., Graves, W. W., & Conant, L. L. (2009). Where is the semantic system? A critical review and meta-analysis of 120 functional neuroimaging studies. *Cerebral Cortex*, *19*, 2767–2796. doi:10.1093/cercor/bhp055

Binder, K. S. & Rayner, K. (1998). Contextual strength does not modulate the subordinate bias effect: Evidence from eye fixations and self-paced reading. *Psychonomic Bulletin and Review*, *5*, 271–276. doi:10.3758/BF03212950

Bird, S., Klein, E., & Loper, E. (2009). *Natural language processing with python*. Sebastopol, CA, USA: O'Reilly Media.

Boyd-Graber, J., Blei, D. M., & Zhu, X. (2007). A topic model for word sense disambiguation. In *Proceedings of the joint conference on empirical methods in natural language processing and computational natural language learning*, June 2007 (pp. 1024–1033). Prague, Czech Republic: Association for Computational Linguistics. Retrieved from http://www.aclweb.org/anthology/D/D07/D07-1109

Bransford, J. D., Barclay, J., & Franks, J. J. (1972). Sentence memory: A constructive versus interpretive approach. *Cognitive Psychology*, *3*, 193–209. doi:10.1016/0010-0285(72)90003-5

Bransford, J. D. & Johnson, M. K. (1972). Contextual prerequisites for understanding: Some investigations of comprehension and recall. *Journal of Verbal Learning and Verbal Behavior*, *11*, 717–726.

Cai, Z. G., Sturt, P., & Pickering, M. J. (2012). The effect of nonadopted analyses on sentence processing. *Language and Cognitive Processes*, *27*, 1286–1311. doi:10.1080/01690965.2011.599657

Chambers, C. G., Tanenhaus, M. K., Eberhard, K. M., Filip, H., & Carlson, G. N. (2002). Circumscribing referential domains during real-time language comprehension. *Journal of Memory and Language*, *47*, 30–49. doi:10.1006/jmla.2001.2832

Christianson, K., Hollingworth, A., Halliwell, J. F., & Ferreira, F. (2001). Thematic roles assigned along the garden path linger. *Cognitive Psychology*, *42*, 368–407. doi:10.1006/cogp.2001.0752

Colbert-Getz, J. & Cook, A. E. (2013). Revisiting effects of contextual strength on the subordinate bias effect: Evidence from eye movements. *Memory and Cognition*, 1–13. doi:10.3758/s13421-013-0328-3

Duffy, S. A., Morris, R. K., & Rayner, K. (1988). Lexical ambiguity and fixation times in reading. *Journal of Memory and Language*, *27*, 429–446. doi:10.1016/0749-596X(88)90066-6

Eberhard, K. M., Spivey-Knowlton, M. J., Sedivy, J. C., & Tanenhaus, M. K. (1995). Eye movements as a window into real-time spoken language comprehension in natural contexts. *Journal of Psycholinguistic Research*, *24*, 409–436. doi:10.1007/BF02143160

Eysenck, M. W. & Keane, M. T. (2010). *Cognitive psychology: A student's handbook* (6th ed.). New York, NY, USA: Psychology Press.

Federmeier, K. D. & Kutas, M. (1999). A rose by any other name: Long-term memory structure and sentence processing. *Journal of Memory and Language*, *41*, 469–495. doi:10.1006/jmla.1999.2660

Ferreira, F. & Clifton, C. (1986). The independence of syntactic processing. *Journal of Memory and Language*, *25*, 348–368. doi:10.1016/0749-596X(86)90006-9

Fischler, I., Bloom, P. A., Childers, D. G., Roucos, S. E., & Perry, N. W. (1983). Brain potentials related to stages of sentence verification. *Psychophysiology*, *20*, 400–409. doi:10.1111/j.1469-8986.1983.tb00920.x

Fodor, J. D. & Frazier, L. (1980). Is the human sentence parsing mechanism an atn? *Cognition*, *8*, 417–459. doi:10.1016/0010-0277(80)90003-7

Fodor, J. D., Ni, W., Crain, S., & Shankweiler, D. (1996). Tasks and timing in the perception of linguistic anomaly. *Journal of Psycholinguistic Research*, *25*, 25–57. doi:10.1007/BF01708419

Fodor, J. A. (1983). *The modularity of mind*. Cambridge, MA, USA: MIT Press.

Forster, K. I., Guerrera, C., & Elliot, L. (2009). The maze task: Measuring forced incremental sentence processing time. *Behavior Research Methods*, *41*, 163–171. doi:10.3758/BRM.41.1.163

Frazier, L. & Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cognition*, *6*, 291–325. doi:10.1016/0010-0277(78)90002-1

Frazier, L. & Rayner, K. (1982). Making and correcting errors during sentence comprehension: Eye movements in the analysis of structurally ambiguous sentences. *Cognitive Psychology*, *14*, 178–210. doi:10.1016/0010-0285(82)90008-1

Friederici, A. D. (1995). The time course of syntactic activation during language processing: A model based on neuropsychological and neurophysiological data. *Brain and Language*, *50*, 259–281. doi:10.1006/brln.1995.1048

Friederici, A. D. (2012). The cortical language circuit: From auditory perception to sentence comprehension. *Trends in Cognitive Sciences*, *16*, 262–268. doi:http://dx.doi.org/10.1016/j.tics.2012.04.001

Friederici, A. D., Pfeifer, E., & Hahne, A. (1993). Event-related brain potentials during natural speech processing: Effects of semantic, morphological and syntactic violations. *Cognitive Brain Research*, *1*, 183–192. doi:10.1016/0926-6410(93)90026-2

Gale, W., Church, K., & Yarowsky, D. (1992). A method for disambiguating word senses in a large corpus. *Computers and the Humanities*, *26*, 415–439.

Gernsbacher, M. A. (1990). *Language comprehension as structure building*. Hillsdale, NJ, USA: Lawrence Erlbaum Associates.

Gernsbacher, M. A., Varner, K. R., & Faust, M. E. (1990). Investigating differences in general comprehension skill. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *16*, 430–445. doi:10.1037/0278-7393.16.3.430

Hagoort, P. & van Berkum, J. J. A. (2007). Beyond the sentence given. *Philosophical Transactions of the Royal Society B: Biological Sciences*, *362*, 801–811. doi:10.1098/rstb.2007.2089

Hambrick, D. Z. & Engle, R. W. (2002). Effects of domain knowledge, working memory capacity, and age on cognitive performance: An investigation of the knowledge-is-power hypothesis. *Cognitive Psychology*, *44*, 339–387. doi:10.1006/cogp.2001.0769

Hare, M., McRae, K., & Elman, J. L. (2003). Sense and structure: Meaning as a determinant of verb subcategorization prefences. *Journal of Memory and Language*, *48*, 281–303. doi:10.1016/S0749-596X(02)00516-8

Huettig, F., Rommers, J., & Meyer, A. S. (2011). Using the visual world paradigm to study language processing: A review and critical evaluation. *Acta Psychologica*, *137*, 151–171. doi:10.1016/j.actpsy.2010.11.003

Hughes, J. & Jacobs, B. (2004). Simulations in coalgebra. *Theoretical Computer Science*, *327*, 71–108. doi:10.1016/j.tcs.2004.07.022

Hyönä, J. & Lorch, R. F. (2004). Effects of topic headings on text processing: Evidence from adult readers' eye fixation patterns. *Learning and Instruction*, *14*, 131–152. doi:10.1016/j.learninstruc.2004.01.001

Jacobs, B. (1995). Objects and classes, co-algebraically. In B. Freitag, C. Jones, C. Lengauer, & H.-J. Schek (Eds.), *Object orientation with parallelism and persistence* (pp. 83–103). Norwell, MA, USA: Kluwer Academic Publishers.

Jacobs, B. (2012). *Introduction to coalgebra. Towards mathematics of states and observations*. Unpublished manuscript, Institute for Computing and Information Sciences, Radboud University. Nijmegen, The Netherlands. Retrieved from http://www.cs.ru.nl/B.Jacobs/CLG/JacobsCoalgebraIntro.pdf

Jacobs, B. & Rutten, J. (2011). An introduction to (co)algebras and (co)induction. In D. Sangiorgi & J. Rutten (Eds.), *Advanced topics in bisimulation and coinduction* (pp. 38–99). Cambridge Tracts in Theoretical Computer Science. Cambridge, UK: Cambridge University Press. doi:10.1017/CBO9780511792588.003

Janssen, T. M. V. (2012). Montague semantics. In E. N. Zalta (Ed.), *The stanford encyclopedia of philosophy* (Winter 2012).

Johnson-Laird, P. N. (1983). *Mental models: Towards a cognitive science of language, inference, and consciousness*. Cambridge, MA, USA: Harvard University Press.

Jurafsky, D. & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, speech recognition, and computational linguistics* (2nd ed.). Pearson Prentice Hall.

Kambe, G., Rayner, K., & Duffy, S. A. (2001). Global context effects on processing lexically ambiguous words: Evidence from eye fixations. *Memory and Cognition*, *29*, 363–372. doi:10.3758/BF03194931

Kay, M. (2005). A life of language. *Computational Linguistics*, *31*, 425–438. doi:10.1162/089120105775299159

King, M. (1992). Epilogue: On the relation between computational linguistics and formal semantics. In *Computational linguistics and formal semantics*. Cambridge, UK: Cambridge University Press.

Kintsch, W. (1988). The role of knowledge in discourse comprehension: A construction-integration model. *Psychological Review*, *95*, 163–182. doi:10.1037/0033-295X.95.2.163

Kizach, J., Nyvad, A. M., & Christensen, K. R. (2013, October). Structure before meaning: Sentence processing, plausibility, and subcategorization. *PLoS ONE*, *8*(10), e76326. doi:10.1371/journal.pone.0076326

Klepousniotou, E. (2002). The processing of lexical ambiguity: Homonymy and polysemy in the mental lexicon. *Brain and Language*, *81*, 205–223. doi:10.1006/brln.2001.2518

Krovetz, R. (1998). More than one sense per discourse. *NEC Princeton NJ Labs., Research Memorandum*.

Kuperberg, G. R. (2007). Neural mechanisms of language comprehension: Challenges to syntax. *Brain Research*, *1146*, 23–49. doi:10.1016/j.brainres.2006.12.063

Kutas, M. & Federmeier, K. D. (2000). Electrophysiology reveals semantic memory use in language comprehension. *Trends in Cognitive Sciences*, *4*, 463–470. doi:10.1016/S1364-6613(00)01560-6

Kutas, M. & Federmeier, K. D. (2009). Thirty years and counting: Finding meaning in the N400 component of the event-related brain potential (ERP). *Annual Review of Psychology*, *62*, 621–647. doi:10.1146/annurev.psych.093008.131123

Kutas, M. & Hillyard, S. A. (1980). Reading senseless sentences: brain potentials reflect semantic incongruity. *Science*, *207*(4427), 203–205. doi:10.1126/science.7350657

Kutas, M. & van Petten, C. (1994). Psycholinguistics electrified: Event-related brain potential investigations. In M. A. Gernsbacher (Ed.), *Handbook of psycholinguistics* (pp. 83–143). San Diego, CA, USA: Academic Press.

Lakoff, G. (1987). *Women, fire, and dangerous things: What categories reveal about the mind*. Chicago, IL, USA: University of Chicago Press.

Lau, E. F., Phillips, C., & Poeppel, D. (2008). A cortical network for semantics: (de)constructing the N400. *Nature Reviews Neuroscience*, *9*, 920–933. doi:10.1038/nrn2532

Lawvere, F. W. & Schanuel, S. H. (2009). *Conceptual mathematics: A first introduction to categories* (2nd ed.). Cambridge, UK: Cambridge University Press.

MacDonald, M. C., Pearlmutter, N. J., & Seidenberg, M. S. (1994). The lexical nature of syntactic ambiguity resolution. *Psychological Review*, *101*, 676–703. doi:10.1037/0033-295X.101.4.676

Magnini, B., Strapparava, C., Pezzulo, G., & Gliozzo, A. (2002). The role of domain information in word sense disambiguation. *Natural Language Engineering*, *8*, 359–373. doi:10.1017/S1351324902003029

Manning, C. D. & Schütze, H. (2001). *Foundations of statistical natural language processing*. Cambridge, MA, USA: MIT Press.

Marslen-Wilson, W. D. (1975). Sentence perception as an interactive parallel process. *Science*, *189*(4198), 226–228. doi:10.1126/science.189.4198.226

McElree, B. & Griffith, T. (1995). Syntactic and thematic processing in sentence comprehension: Evidence for a temporal dissociation. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *21*, 134–157. doi:10.1037/0278-7393.21.1.134

Metusalem, R., Kutas, M., Urbach, T. P., Hare, M., McRae, K., & Elman, J. L. (2012). Generalized event knowledge activation during online sentence comprehension. *Journal of Memory and Language*, *66*, 545–567. doi:10.1016/j.jml.2012.01.001

Miller, S., Stallard, D., Bobrow, R., & Schwartz, R. (1996). A fully statistical approach to natural language interfaces. In *Proceedings of the 34th annual meeting on association for computational linguistics* (pp. 55–61). ACL '96. Santa Cruz, California: Association for Computational Linguistics. doi:10.3115/981863.981871

Morris, R. K. (2006). Lexical processing and sentence context effects. In M. J. Traxler & M. A. Gernsbacher (Eds.), *Handbook of psycholinguistics* (2nd ed., pp. 377–401). London: Academic Press. doi:10.1016/B978-012369374-7/50011-0

Navigili, R. (2009). Word sense disambiguation: A survey. *ACM Computing Surveys*, *41*(2), 10:1–10:69. doi:10.1145/1459352.1459355

Nieuwland, M. S. & van Berkum, J. J. A. (2006). When peanuts fall in love: N400 evidence for the power of discourse. *Journal of Cognitive Neuroscience*, *18*, 1098–1111. doi:10.1162/jocn.2006.18.7.1098

Nivre, J. (2008). Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, *34*, 513–553. doi:10.1162/coli.07-056-R1-07-027

Novichkova, S., Egorov, S., & Daraselia, N. (2003). Medscan, a natural language processing engine for MEDLINE abstracts. *Bioinformatics*, *19*, 1699–1706. doi:10.1093/bioinformatics/btg207

Otten, M. & van Berkum, J. J. A. (2007). What makes a discourse constraining? comparing the effects of discourse message and scenario fit on the discourse-dependent N400 effect. *Brain Research*, *1153*, 166–177. doi:10.1016/j.brainres.2007.03.058

Patson, N. D., Darowski, E. S., Moon, N., & Ferreira, F. (2009). Lingering misinterpretations in garden-path sentences: Evidence from a paraphrasing task. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *35*, 280–285. doi:10.1037/a0014276

Peelle, J. E., Cooke, A., Moore, P., Vesely, L., & Grossman, M. (2007). Syntactic and thematic components of sentence processing in progressive nonfluent aphasia and nonaphasic frontotemporal dementia. *Journal of Neurolinguistics*, *20*, 482–494. doi:10.1016/j.jneuroling.2007.04.002

Pickering, M. J. & Frisson, S. (2001). Processing ambiguous verbs: Evidence from eye movements. *Journal of Experimental Psychology: Learning, Memory and Cognition*, *27*, 556–573. doi:10.1037//0278-7393.27.2.556

Pickering, M. J. & van Gompel, R. P. G. (2006). Syntactic parsing. In M. J. Traxler & M. A. Gernsbacher (Eds.), *Handbook of psycholinguistics* (2nd ed., pp. 455–503). London, UK: Academic Press. doi:10.1016/B978-012369374-7/50013-4

Radvansky, G. A. & Dijkstra, K. (2007). Aging and situation model processing. *Psychonomic Bulletin and Review*, *14*, 1027–1042. doi:10.3758/BF03193088

Resnik, P. (1993). *Selection and information: A class-based approach to lexical relationships* (Doctoral dissertation, Department of Computer and Information Science, University of Pennsylvania). Retrieved from http://umiacs.umd.edu/~resnik/pubs/resnik_dissertation.pdf

Roche, E. & Schabes, Y. (1997). Introduction. In E. Roche & Y. Schabes (Eds.), *Finite-state language processing* (pp. 1–66). Cambridge, MA, USA: MIT Press.

Rogers, T. T., Hocking, J., Noppeney, U., Mechelli, A., Gorno-Tempini, M. L., Patterson, K., & Price, C. J. (2006). Anterior temporal cortex and semantic memory: Reconciling findings from neuropsychology and functional imaging. *Cognitive, Affective, and Behavioral Neuroscience*, *6*, 201–213. doi:10.3758/CABN.6.3.201

Rosen, K. H. (2007). *Discrete mathematics and its applications* (6th ed.). McGraw-Hill.

Rutten, J. (2000). Universal coalgebra: A theory of systems. *Theoretical Computer Science*, *249*. doi:10.1016/S0304-3975(00)00056-6

Schneider, D. & Phillips, C. (2001). Grammatical search and reanalysis. *Journal of Memory and Language*, *45*, 308–336. doi:10.1006/jmla.2001.2777

Schneider, W., Körkel, J., & Weinert, F. E. (1989). Domain-specific knowledge and memory performance: A comparison of high- and low-aptitude children. *Journal of Educational Psychology*, *81*, 306–312. doi:10.1037/0022-0663.81.3.306

Seidenberg, M. S., Tanenhaus, M. K., Leiman, J. M., & Bienkowski, M. (1982). Automatic access of the meanings of ambiguous words in context: Some limitations of knowledge-based processing. *Cognitive Psychology*, *14*, 489–537. doi:10.1016/0010-0285(82)90017-2

Sereno, S. C., O'Donnell, P. J., & Rayner, K. (2006). Eye movements and lexical ambiguity resolution: Investigating the subordinate-bias effect. *Journal of Experimental Psychology: Human Perception and Performance*, *32*, 335–350. doi:10.1037/0096-1523.32.2.335

Simone, N. (1965). Sinnerman. On *Pastel blues*. Philips Records.

Simpson, G. B. & Burgess, C. (1985). Activation and selection processes in the recognition of ambiguous words. *Journal of Experimental Psychology: Human Perception and Performance*, *11*, 28–39. doi:10.1037/0096-1523.11.1.28

Snedeker, J. & Trueswell, J. C. (2004). The developing constraints on parsing decisions: The role of lexical-biases and referential scenes in child and adult sentence processing. *Cognitive Psychology*, *49*, 238–299. doi:http://dx.doi.org/10.1016/j.cogpsych.2004.03.001

Spilich, G. J., Vesonder, G. T., Chiesi, H. L., & Voss, J. F. (1979). Text processing of domain-related information for individuals with high and low domain knowledge. *Journal of Verbal Learning and Verbal Behavior*, *18*, 275–290. doi:10.1016/S0022-5371(79)90155-5

Staub, A. (2007). The parser doesn't ignore intransitivity, after all. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *33*, 550–569. doi:10.1037/0278-7393.33.3.550

Sturt, P. (2007). Semantic re-interpretation and garden path recovery. *Cognition*, *105*, 477–488. doi:10.1016/j.cognition.2006.10.009

Sturt, P., Pickering, M. J., Scheepers, C., & Crocker, M. W. (2001). The preservation of structure in language comprehension: Is reanalysis the last resort. *Journal of Memory and Language*, *45*, 283–301. doi:10.006/jmla.2001.2776

Tomasello, M. (1988). The role of joint attentional processes in early language development. *Language Sciences*, *10*, 69–88. doi:10.1016/0388-0001(88)90006-X

Trueswell, J. C., Sekerina, I., Hill, N. M., & Logrip, M. L. (1999). The kindergarten-path effect: Studying on-line sentence processing in young children. *Cognition*, *73*, 89–134. doi:http://dx.doi.org/10.1016/S0010-0277(99)00032-3

Turing, A. M. (1950). I.—computing machinery and intelligence. *Mind*, *59*(236), 433–460. doi:10.1093/mind/LIX.236.433

Twilley, L. C., Dixon, P., Taylor, D., & Clark, K. (1994). University of alberta norms of relative meaning frequency for 566 homographs. *Memory and Cognition*, *22*, 111–126. doi:10.3758/BF03202766

van Berkum, J. J. A., Hagoort, P., & Brown, C. M. (1999). Semantic integration in sentences and discourse: Evidence from the N400. *Journal of Cognitive Neuroscience*, *11*(6), 657–671. doi:10.1162/089892999563724

van den Bosch, A. & Buchholz, S. (2002). Shallow parsing on the basis of words only: A case study. In *Proceedings of the 40th meeting of the association for computational linguistics* (pp. 433–440).

van der Schoot, M., Vasbinder, A. L., Horsley, T. M., Reijntjes, A., & van Lieshout, E. C. (2009). Lexical ambiguity resolution in good and poor comprehenders: An eye fixation and self-paced reading study in primary school children. *Journal of Educational Psychology*, *101*, 21–36. doi:10.1037/a0013382

van Dijk, T. A. & Kintsch, W. (1983). *Strategies of discourse comprehension*. New York, NY, USA: Academic Press.

van Petten, C. & Kutas, M. (1991). Influences of semantic and syntactic context on open- and closed-class words. *Memory and Cognition*, *19*, 95–112. doi:10.3758/BF03198500

van Gompel, R. P., Pickering, M. J., Pearson, J., & Jacob, G. (2006). The activation of inappropriate analyses in garden-path sentences: Evidence from structural priming. *Journal of Memory and Language*, *55*, 335–362. doi:10.1016/j.jml.2006.06.004

van Gompel, R. (2006). Sentence processing. In K. Brown (Ed.), *Encyclopedia of language and linguistics* (2nd ed., pp. 251–255). Oxford, UK: Elsevier. doi:10.1016/B0-08-044854-2/00789-6

Ward, J. (2010). *Student's guide to cognitive neuroscience* (2nd ed.). Hove, UK: Psychology Press.

Weighall, A. R. (2008). The kindergarten path effect revisited: Children's use of context in processing structural ambiguities. *Journal of Experimental Child Psychology*, *99*, 75–95. doi:http://dx.doi.org/10.1016/j.jecp.2007.10.004

Wiley, J. & Rayner, K. (2000). Effects of titles on the processing of text and lexically ambiguous words: Evidence from eye movements. *Memory and Cognition*, *28*, 1011–1021. doi:10.3758/BF03209349

Wolter, U. (2012). *Category theory and diagrammatic modelling*. Unpublished manuscript, Department of Informatics, University of Bergen. Bergen, Norway.

Zwaan, R. A. (1999). Situation models: The mental leap into imagined worlds. *Current Directions in Psychological Science*, *8*, 15–18.