# Cognitive Models of Syntax and Sentence Processing

Vera Demberg & Frank Keller

## 1. Introduction

### *1.1. Motivation*

Cognitive models of sentence processing implement cognitive processing theories. They should hence display the same processing difficulties as humans and be able to predict processing ease where humans experience processing ease. This is usually not the case for general computational natural language processing models, which may behave very differently from humans in general in terms of when processing difficulties occur. For instance, a parser may not show difficulties for sentences where humans are garden-pathed, or may correctly process center embedding constructions which humans find difficult, while having a lot of trouble in recovering the intended structure of sentences containing minor ungrammaticalities (which is usually not difficult for humans), or having difficulty in dealing with ambiguities which pose little problems for humans.

Computational cognitive models are useful for cognitive science because they spell out all necessary assumptions of a theory explicitly, and can show effects of complex interactions of different aspects of a processing theory. Cognitive models can also be a tool for identifying what additional assumptions would have to be made to implement a theory. An implemented model makes a theory testable empirically, including not only statements about which construction should be easier or more difficult than which other construction, but also providing an estimate for the *size* of an effect. It can also generate quantitative predictions for so-far untested constructions, which can in turn be tested empirically to provide additional support for the model or contradict the model.

Cognitive models can hence give us insight into cognition, by attempting to give an explanation for the phenomena that we observe, or by at least providing an elegant description of these phenomena. They also have potential application in wide areas of NLP, if systems could get closer to human-levels of robustness in dealing with noise and ungrammatical input. Further applications for which cognitive modeling is highly relevant include readability assessment, and dialog systems: more natural language-based human-computer interaction could be achieved if dialog systems can accurately model human processing and can thus adapt the utterances generated by the system more optimally to human comprehension.

A computational cognitive model typically consists of an architecture which specifies the representations used by the computational model (see Section 2.1), a processing algorithm (see Section 2.2) and a linking theory (see Section 4). Applied to models of human sentence processing, the linking theory specifies how the parsing operations of the algorithm are connected to processing difficulty effects such as longer reading times, certain types of ERP effects or changes in acceptability.

In this chapter, we will focus on mostly syntactic aspects of sentence processing, an area which has received a lot of attention in the human language processing community. Note that a comprehensive answer to our modeling challenge would also include phonological and morphological processing, semantic inference, discourse processing, and other non-syntactic aspects of language processing. Furthermore, established results regarding the interface between language processing and non-linguistic cognition (e.g., the sensorimotor system) should ultimately be accounted for in a model of human language cognition.

### *1.2. Key properties*

The central tenet of computational modeling in psycholinguistics is to capture key properties of human language processing, as established by psycholinguistic experimentation. A striking property of the human language processor is its *efficiency and robustness*. For the vast majority of sentences, it will effortlessly and rapidly deliver the correct analysis, even in the face of noise and ungrammaticalities. There is considerable experimental evidence that shallow processing strategies are used to achieve this. The processor also achieves *broad coverage*: it can deal with a wide variety of syntactic constructions, and is not restricted by the domain, register, or modality of the input.

Evidence from psycholinguistic research shows that human language comprehension is *incremental*. Comprehenders do not wait until the end of the sentence before they build a syntactic representation for the sentence; rather, they construct a sequence of partial representations for sentence prefixes. (A sentence prefix is the subsequence of a sentence from the first word to the word currently being processed.) Experimental results indicate that each new word that is read or heard triggers an update of the representation constructed so far (Konieczny, 2000; Tanenhaus, Spivey-Knowlton, Eberhard, & Sedivy, 1995). There is also evidence for connectedness in human language processing (Sturt & Lombardo, 2005). *Connectedness* means that all input words are attached to the same syntactic structure (though connected structures can be constructed in parallel); comprehenders build no unconnected tree fragments, even for the sentence prefixes that arise during incremental processing.

Furthermore, the processor is able to make *predictions* about upcoming material on the basis of sentence prefixes. Evidence for prediction comes, for example, from the finding that people are able to anticipate the argument of a verb (increased fixations on the argument in a visual world paradigm,[1] even before this argument is heard, (Kamide, Scheepers, & Altmann, 2003). Additional evidence for prediction is provided by experiments in which N400 effects are observed when the form of the determiner does not match the anticipated noun (Delong, Urbach, & Kutas, 2005), or the processing of *either . . . or* constructions, where the word *or* and a following noun phrase are read faster in contexts that include *either* (Staub & Clifton, 2006). This effect can be explained if we assume that or and the second conjunct are predicted when the parser reaches *either*.

Another key property of human language processing is the fact that it operates with limited memory, and that structures in memory are subject to decay and interference. In particular, the processor is known to incur a distance-based *memory cost*: combining the head of a phrase with its syntactic dependents is more difficult the more dependents have to be integrated and the further away they are. This integration process is also subject to interference from similar items that have to be held in memory at the same time.

## 2. Formalisms and algorithms

### *2.1. Formalisms*

#### 2.1.1. Context-free grammar

In order to build a model of human sentence processing, we have to make assumptions about how sentence structure is formalized and processed. Oftentimes, the psycholinguistic literature has couched these assumptions in the form of a context-free grammar (CFG). Context-free grammars model phrase structure, i.e., the hierarchical relationships that hold between the

[1] In the visual world paradigm, participants see a visual scene while hearing a sentence. Their eye-movements on the scene in reaction to the sentence are recorded, providing information about the current state of their sentence processing.

different elements of a sentence. A CFG typically decomposes a sentence into phrasal categories (e.g., S for sentence, NP for noun phrase, VP for verb phrase) and lexical categories (parts of speech) such as determiner (Det), noun (N), and verb (V). These categories are then combined into context-free grammar rules, such as the following:

[Figure 22.1 here]

| | | | | | | |
|---|---|---|---|---|---|---|
| (1) | S | → | NP VP NP | Det | → | The |
| | NP | → | Det N | N | → | Kitten |
| | VP | → | V NP | N | → | Dog |
| | | | | V | → | Bit |

A *derivation* is the sequence that results from applying a sequence of context-free rules, starting from a start symbol (here S). As an example, consider the derivation for the sentence *the kitten bit the dog* relative to the grammar in (1):

(2) S ⇒ NP VP ⇒ NP V NP ⇒ NP V Det N ⇒ NP bit Det N ⇒ NP bit Det dog ⇒ NP bit the dog ⇒ Det N bit the dog ⇒ the N bit the dog ⇒ the kitten bit the dog

A derivation assigns a *syntax tree* to a sentence. The syntax tree notates which rules have been applied in the derivation (but not the order in which they have been applied). An example is Figure 22.1, which gives the syntax tree for the derivation in (2).

Crucially, a sentence can have multiple syntax trees, each of which can have a different interpretation, a phenomenon called syntactic ambiguity. The following sentences are syntactically ambiguous:[2]

(3) a. She sat on the chair covered in dust.

b. I put the book on the table in the kitchen.

c. Milk drinkers are turning to powder.

d. Old school pillars are replaced by alumni.

In all of these cases, the whole sentence is ambiguous, which is referred to as *global ambiguity*.

As the human sentence processor builds the syntax tree for the input incrementally (see Section 1.2), it can encounter an ambiguity that disappears later when more of the input is read. This is referred to as *local ambiguity*. An example for a local ambiguity is:

(4) The athlete realized his potential . . .

a. . . . at the competition.

b. . . . could make him a world-class sprinter.

[Figure 22.2 here]

The local syntactic trees for (4a) and (4b) are given in Figure 22.2.

Both trees are compatible with the input up until *potential*; only the next word disambiguates; however, the processor commits to a single (wrong) tree early on, and trips up

[2] Examples from http://www.fun-with-words.com/ambiguous_garden_path.html.

when later input is inconsistent with that tree; presumably, the processor now has to compute a new tree that is consistent with the input, which results in longer reading times, reverse eye-movements, lower comprehension accuracies, etc. (we will come back to this in Section 3.1).

### 2.1.2. Tree-Adjoining Grammar

Tree-Adjoining Grammar (TAG, Joshi, Levy, & Takahashi, 1975) is an extension of context-free grammar that assumes that a grammar consists of lexicalized tree fragments rather than context-free rules. TAG assumes two operations that can combine elementary trees: substitution combines an initial tree with a partial tree (this happens at substitution nodes, marked with "↓"); adjunction combines an auxiliary tree with a partial tree (this happens at adjunction nodes, marked with "∗"). A TAG derivation consists of a sequence of substitution and adjunction operations, applied to the initial and auxiliary trees associated with the words in the sentence.

Standard TAG derivations are not incremental, i.e., words are not processed in the left-to-right order of the input. As we saw in Section 1.2, incrementality is a key property of the human sentence processor, which means that TAG needs to be extended in order to be utilized in a model of human sentence processing. Such an extension has been proposed in the form of Psycholinguistically Motivated Tree-adjoining Grammar (PLTAG, Demberg & Keller, 2008b). PLTAG parsing proceeds strictly incrementally; all words in the input are always connected under a single syntactic node. This strict incrementality is made possible through a *prediction* mechanism which allows the parser to insert syntactic structure into the unfolding representation before its syntactic head has been encountered in the input. For example, when observing the noun phrase *John* at the beginning of a sentence, the parser would predict that this NP might be the subject of a sentence and would predict that a verb phrase will be encountered. (In standard

TAG by contrast, the verb phrase and role of the NP in the sentence would only be constructed once the verb is actually encountered in the input.) Figure 22.3 illustrates how this prediction mechanism allows PLTAG to connect the words *Peter* and *often* before the verb *sleeps* is encountered. The prediction mechanism makes use of so-called prediction trees, a new type of elementary tree that carries indices on each node (which mark their status as predicted and not yet grounded in input), as shown in Figure 22.3d. Note that prediction trees (as opposed to standard TAG trees) need not contain a lexical anchor.

The prediction mechanism is complemented by a *verification* mechanism that matches predictions against subsequent input, and when matching removes the prediction markers to indicate that the previously predicted structure is now grounded in actually observed material (the structure that comes with the verb *sleeps* in our example above).

[Figure 22.3 here]

PLTAG is a fully implemented model that consists of an algorithm for extracting a PLTAG grammar from an annotated corpus, a parsing algorithm that processes PLTAG structures incrementally, and a probability model over PLTAG derivations. We will return to this in Section 6.3.

### 2.1.3. Dependency grammar

Traditional context-free grammars capture the *configurational structure* of sentences, i.e., they describe which phrases a sentences is composed of and how these phrases are ordered relative to each other, both sequentially and hierarchically. However, this view of syntax has been challenged, as it makes it difficult to analyze languages that allow discontinuous constituents or have free or flexible word order.

Dependency grammar is one of the answers to this challenge. It is a formalism that starts from the assumption that the grammatical relations between words, rather than their configurations, are key to sentence structure. For instance, the important thing about verbs such as *like* is that they license two NPs: an agent, found in subject position or with nominative inflection, and a patient, found in object position or with accusative inflection. Which arguments are licensed, and which roles they play, depends on the verb. The configuration of these arguments is secondary.

Dependency grammar therefore models the grammatical relations that hold between words; the claim is that these can be identified even in non-configurational languages. A dependency structure consists of *dependency relations*, which are binary and asymmetric. A relation is a tuple of a *head h*, a dependent *d*, and a *label*, which identifies the relation between *h* and *d*. An example for a dependency structure can be found in Figure 22.4.

[Figure 22.4 here]

[Figure 22.5 here]

Formally, a dependency structure of a sentence is a graph with the words of the sentence as its nodes, linked by directed, labeled edges, with the following properties: (a) connectedness: every node is related to at least one other node, and (through transitivity) to ROOT; (b) single headedness: every node (except root) has exactly one incoming edge (from its head); (c) acyclicity: the graph cannot contain cycles of directed edges.

These conditions ensure that the dependency structure is a tree. We distinguish projective and non-projective dependency trees: A dependency tree is *projective* with respect to a particular linear order of its nodes if, for all edges $h \rightarrow d$ and nodes $w$, $w$ occurs between $h$ and $d$ in linear order only if $w$ is dominated by $h$ (i.e., in the transitive closure of the arc relation). A dependency tree is *non-projective* if $w$ can occur between $h$ and $d$ in linear order without being dominated by $h$. This is illustrated in Figure 22.5.

Dependency grammar is the formalism that underlies Dependency Locality Theory, one of the key approaches to psycholinguistic modeling, which will be discussed in Section 6.1.

### *2.2. Parsing algorithms*

A parsing algorithm takes a grammar and a sentence and derives a syntax tree for the sentence. A range of parsing algorithms have been proposed in the literature (for an overview, see Jurafsky & Martin, 2009). The most basic distinction is between top-down and bottom-up parsing algorithms. A top-down parser starts with the start symbol of the grammar, and expands each phrasal category until it reaches the leaf nodes (the words of the sentence). We already encountered an example for a top-down derivation in (2) for the sentence *the kitten bit the dog* with respect to the grammar in (1). A bottom-up parser starts with the leaf nodes and replaces each of them with phrasal categories until it reaches the start symbol. The bottom-up derivation for the same sentence and grammar is as follows:

(5) the kitten bit the dog ⇒ Det kitten bit the dog ⇒ Det N bit the dog ⇒ Det N V the dog ⇒ Det N V Det dog ⇒ Det N V Det N ⇒ NP V Det N ⇒ NP V NP ⇒ NP VP ⇒ S

As we saw in Section 2.1.1, a derivation for a sentence using a context-free grammar can be ambiguous, i.e., it can assign more than one syntax tree to a given sentence, either locally or globally. Ambiguity occurs for every realistically large grammar, which typically has thousands of rules. This means that naive top-down and bottom-up parsers have a large search space to explore: for example, in (2), there could be many alternative ways of expanding S or NP; in (5), there could be multiple ways of expanding *bit* or *dog*, each allowing to derive multiple phrasal categories.

The standard solution to this problem is to employ a *chart*, a data structure that stores subtrees for each phrase the parser has discovered. These subtrees can then be used in subsequent analyses that require this phrase, without having to be constructed again, making parsing much more efficient. Standard examples for chart parsing algorithms are the CKY algorithm and the Earley algorithm (Jurafsky & Martin, 2009). Of particular interest from a psycholinguistic point of view is left-corner parsing, a chart-parsing strategy that combines top-down and bottom-up information, and has been argued to be cognitively plausible based on its memory requirements (Abney & Johnson, 1991). While we cannot give a formal definition of the left-corner algorithm here, Figure 22.6 shows an example of a left-corner derivation (again for *the kitten bit the dog* with respect to (1)).

The key idea is that the parser starts bottom up with *the*, but then makes a top-down prediction of a rule whose left corner is Det, i.e., the rule NP → Det N. Next, it completes the NP bottom up (by parsing *kitten*), which allows it to make another top-down prediction, as NP is the left corner of the rule S → NP VP. Top-down and bottom-up predictions alternate until the input is consumed.

All the parsing algorithms we discussed so far are designed for CFG, but they can be adapted for TAG (Joshi & Schabes, 1997). For dependency grammars, a different set of parsing algorithms is typically employed, including transition-based parsing (Nivre, 2003) and the maximum spanning tree algorithm (McDonald, Pereira, Ribarov, & Hajič, 2005). An early implemented broad-coverage incremental probabilistic processing model was suggested by Crocker and Brants (2000). Their model did not claim cognitive plausibility based on the architecture (cascaded Markov models) but through its broad-coverage incremental parsing process.

## 3. Specific constructions

### *3.1. Garden paths*

Garden path sentences are one of the classic human language processing phenomena. A *garden path effect* refers to a situation where a language processing difficulty becomes so severe that the reader becomes aware of the difficulty of the sentence. The reader is initially “lured” into a very probable interpretation, which later turns out to be incompatible with the end of the sentence. The difficulty of reanalyzing the sentences can be so strong that correct analysis cannot be recovered, and the sentence is judged to be ungrammatical.

The most famous example is probably the sentence of Bever (1970) shown in (6):

(6) The horse raced past the barn fell.

The reader initially analyses *raced* as a simple past form and hence the main verb of the sentence. However, this is incompatible with *fell*. For the correct interpretation of the sentence, *raced past the barn* must be analyzed as a reduced relative clause, and *raced* hence as a past participle.

[Figure 22.6 here]

Garden path effects are usually not only caused by difficult syntactic constructions, but are also dependent on semantics, i.e., they often consist of syntactically slightly difficult structures that in addition are made implausible by the semantics of the sentence.

There is a qualitative difference between a minor processing difficulty which people often do not become aware of, and complete processing break-down, where some people are not able to recover the correct analysis at all. Parallel models explain this difference by assuming that the processor only focuses on a finite number of most probable analyses, and that the correct analysis has fallen out of the reader's search beam[3] because it was highly unlikely compared to alternative analyses. Hence, the sentence must be re-analyzed from scratch, which is only successful if enough memory resources can be made available for the larger beam needed to process the sentence (Altmann & Steedman, 1988; Gibson, 1991).

### *3.2. Center embedding*

A well-known construction is *center embedding*, where several (of the same type of) structures nested inside one another make sentences increasingly difficult to process (e.g., Eady & Fodor,

[3] The analyses in the search beam are the ones that the processor is actively pursuing. Typically a probability threshold is used to determine which analyses are removed from the beam (pruned).

1981; Chomsky, 1957; Miller & Isard, 1964). Consider the sentences from Gibson (1998) in example (7):

(7) a. The intern [who the nurse supervised] had bothered the administrator [who lost the medical reports].

b. The administrator [who the intern [who the nurse supervised] had bothered] lost the medical reports.

Sentence (7b) has been shown to be considerably more difficult to process than (7a) based on complexity ratings. The difficulty with such sentences is usually modeled via memory access, i.e., as arising from having to keep a large number of similar open structures accessible in memory (Gibson, 1998; Warren & Gibson, 2002), or more specific constraints on the number of clauses that can be parsed at the same time (Kimball, 1973; Lewis, 1996; Stabler, 1994).

### *3.3. Locality and anti-locality*

The term *locality effects* is used to refer to increased processing difficulty (measured typically in terms of increased reading times) for the second element of a long distance dependency. Longer processing times are usually observed for longer distance dependencies, and are essentially a more general notion of center embedding. Locality effects are one possible explanation for the subject relative clause vs. object relative clause asymmetry (i.e., the fact that object relative clauses are harder to process than subject relative clauses). While subject and object relative clauses have the same number of embeddings, an object relative clause has longer dependencies.

Anti-locality effects have been shown for a number of languages, including German (Konieczny, 2000; Konieczny & Döring, 2003) and Hindi (Vasishth & Lewis, 2006). Anti-locality effects refer to the finding that reading times can be shorter at the head when intervening material is inserted between the head and its arguments. Examples for anti-locality effects in German include results presented by Konieczny (2000) and Konieczny and Döring (2003). The experiment reported by Konieczny (2000) found that the verb in verb-final constructions in German is read faster when more material (just one argument, vs. an additional PP vs. a longer, modified PP) occurred before the verb. This finding is contrary to the locality effect found in English center embedding and the subject/object relative clause asymmetry. A similar experiment was conducted by Konieczny and Döring (2003), who also controlled for length of the intervening material between conditions. An example of their materials is shown in (8).

(8) a. Die Einsicht, dass [NP-NOM der Freund] [NP-DAT dem Kunden] [NP-ACC das Auto aus Plastik] verkaufte . . .

*the insight, that the friend the client the car from plastic sold, . . .*

“The insight that the friend sold the client the plastic car . . . ”

b. Die Einsicht, dass [NP-NOM der Freund [NP-GEN des Kunden]] [NP-ACC das Auto aus Plastik] verkaufte, . . .

*the insight, that the friend of the client the car from plastic sold, . . .*

“The insight that the friend of the client sold the plastic car . . . ”

In materials following the example in (8), reading times on the verb *verkaufte* are shorter in (8a) than in (8b), even though the length of interfering elements is exactly identical. Such anti-locality effects can be accounted for in terms of entropy or surprisal, see Section 4.

### *3.4. Local coherence*

It has been observed that processing difficulty can sometimes occur in sentences that are neither ambiguous nor particularly complex (Tabor, Galantucci, & Richardson, 2004). An example for this is the sentence in (9a), which has the same syntactic complexity as (9c).

(9) a. The coach smiled at the player tossed a frisbee by . . .

b. The coach smiled at the player who was tossed a frisbee by . . .

c. The coach smiled at the player thrown a frisbee by . . .

d. The coach smiled at the player who was thrown a frisbee by . . .

The important difference between the sentences is that in (9a) the word sequence *the player tossed a frisbee* is a coherent string of words where *the player* would be the subject of a main verb *tossed*, while *the player thrown a frisbee* cannot be interpreted as such. While (9a) can be expected to be the most difficult sentence among the sentences in (9), because reduced relative clauses are more difficult than non-reduced relative clauses, and ambiguous verb forms are more difficult than unambiguous verb forms, the observed difficulty effect was stronger than would be expected by adding the verb ambiguity and reduced relative clause effects.

A common explanation for the effect is that the locally coherent interpretation of *the player tossed a frisbee* interferes with the globally coherent analysis of the sentence, and has therefore been argued to provide evidence against a view of strictly incremental processing, as the locally coherent analysis should not be calculated in the first place, because *the player* already has a different function in the sentence, and cannot possibly be the subject of *tossed*, and *tossed* cannot be the main verb of the sentence, as there is already a main verb, *smiled*. Alternatively, this type of effect could be explained by a noisy-channel model, where readers maintain uncertainty about what they have processed so far (under this account, readers take into consideration that they may have misheard or misread, or that the speaker may have mispronounced / misspelled). The reader would then generate hypotheses compatible with plausible input that differs from the actually perceived input (Levy, Bicknell, Slattery, & Rayner, 2009). This kind of explanation would still be compatible with rational incremental language processing.

### *3.5. Competition*

*Competition effects* are used to refer to the observation that ambiguous words or structures cause higher processing difficulty than unambiguous words or structures. Such effects were initially observed for lexical ambiguities, in experiments that compared reading times on words that were unambiguous, ambiguous with one predominant meaning or ambiguous with two meanings being similarly prominent. Experimental results showed that fixation times on words with two similarly prevalent meanings were longer than on the other two classes (Duffy, Morris, & Rayner, 1988), which was then explained as both analyses "competing" with one another and hence taking longer to be processed. MacDonald, Pearlmutter, and Seidenberg (1994) suggested that syntactic ambiguities might show similar effects, with supporting evidence provided by experiments on prepositional phrase attachment ambiguities (Spivey-Knowlton & Sedivy, 1995).

### *3.6. Facilitating Ambiguity*

In the so-called *facilitating ambiguity* effect (Traxler, Pickering, & Clifton, 1998), reading times can be faster (as opposed to slower, which would be in line with competition effects) under some circumstances in an ambiguous region compared to an unambiguous region. Consider example (10): the reflexive pronoun *herself/himself* in (10a) and (10b) is unambiguous in that it can only refer to the daughter/colonel respectively. In sentence (10c) however, *himself* is ambiguous as to whether it refers to the son or the colonel.

(10) a. The daughter$_i$ of the colonel$_j$ who shot herself$_{i/*j}$ on the balcony had been very depressed.

b. The daughter$_i$ of the colonel$_j$ who shot himself$_{*i/j}$ on the balcony had been very depressed.

c. The son$_i$ of the colonel$_j$ who shot himself$_{i/j}$ on the balcony had been very depressed.

Reading times were found to be faster on the *himself/herself* and immediately following region in the ambiguous case (1c). This finding is difficult to explain for models who assume competition. Models that can account for facilitating ambiguity effects model the effect by avoiding to fully disambiguate these sentences until necessary.

### *3.7. Commitment and digging-in*

Digging-in effects refer to the finding that a wrong syntactic analysis becomes harder and harder to reanalyze the longer the ambiguous region is. As an example, consider the sentences in (11). Sentences (11a) and (11b) are initially ambiguous at the NP *the book* with respect to whether the NP is an argument of the verb *write* or the subject of the main phrase, while sentences (11c) and (11d) are not (because the verb already has an argument, *the essay*). Subjects initially interpret *the book* as an object of *write* because it is a semantically very likely object of *write* and because *write* is more often seen as a transitive verb than as an intransitive one.

(11) a. As the author wrote the book grew. (ambiguous, short)

b. As the author wrote the book describing Babylon grew. (ambiguous, long)

c. As the author wrote the essay the book grew. (unambiguous, short)

d. As the author wrote the essay the book describing Babylon grew. (unambiguous, long)

It has been shown using acceptability judgments (Ferreira & Henderson, 1991) and reading times (Tabor & Hutchins, 2004), that (11b) is much more difficult than (11a) and the control condition (11d). One would expect (11b) to be the most difficult condition anyway, as it is more complex than (11b) and more ambiguous than (11d). However, Tabor and Hutchins found a difficulty effect on the last word of (11b) (an interaction between length and ambiguity before encountering grew) that goes beyond the main effects of ambiguity and complexity. This effect may be related to the even stronger garden path effects where recovery of the correct analysis may fail completely, or to effects related to the strength of context constraint.

Context constraint has been shown to facilitate processing of an expected word. However, there are also effects indicating that there may be additional effects related to the overall level of constraint of a context on processing. Federmeier, Wlotko, De Ochoa-Dewald, and Kutas (2007) show in an ERP study that while the N400 related to predictability is the same in weakly and strongly constraining contexts, strongly constraining contexts show a larger late positivity.

## 4. Linking theories

We have now seen a range of different grammar formalisms that can describe syntactic structure, as well as different parsing algorithms that describe how to use a grammar to construct the syntactic representation for linguistic input. The function of a so-called “linking theory” or “linking hypothesis” is to suggest what aspects of the parsing process are effortful and hence lead to observable processing difficulty. The linking theory, together with a grammar and a parsing algorithm constitutes a *processing model* (see Section 6), which we can then test as to whether it can account for the data, i.e., both broad-coverage processing and experimental data as outlined above (see Section 3).

### *4.1. Early approaches*

Early attempts at accounting for the occurrence of processing difficulty (with a specific focus on garden path sentences) suggested that language processing may be subject to a range of constraints or preferences that are used to rank alternative parses for a sentence, such that some structures would be predicted to be preferred over others. Processing difficulty would be related to cases where a dispreferred analysis would turn out to be the only viable one. Very strongly dispreferred structures are pruned, which provides a mechanism to account for garden path sentences (cases of extreme processing difficulty). The most important constraints were based on locality preferences (e.g., Right Association, Kimball, 1973; Local Association, Fodor & Frazier, 1978; Late Closure, Frazier, 1979; Final Arguments, Ford, Bresnan, & Kaplan, 1982; Attach Low and Parallel, Hobbs & Bear, 1990; and the Recency Preference, Gibson, 1991). These hypotheses were only descriptive and not very generalizable, and have since been replaced by linking hypotheses that can apply to a larger range of structures and phenomena.

### *4.2. Memory decay and interference*

Based on the observed processing difficulty that humans encounter in sentences with center embeddings or long dependencies (locality effects), several linking theories have been suggested which attribute processing difficulty to retrieving material that has already been processed from memory to integrate it with new upcoming information. Variants of these ideas also take into account whether there are strong competitors, i.e., the degree to which other entities in memory may lead to interference effects when retrieving the relevant entities. The most well-known linking theory that describes this aspect is the integration cost component of Dependency Locality Theory (DLT; Gibson, 1998, 2000). Processing difficulty is suggested to be related to the distance between a dependent and its head: in order to semantically integrate the two concepts, the dependent needs to be retrieved from memory. Different ways of quantifying the distance have been suggested in the literature, and include counting the number of new discourse referents that have been encoded since processing the dependent (Gibson, 1998, p. 12f; Warren & Gibson, 2002), simply counting intervening words (Demberg, Keller, & Koller, 2013; Temperley, 2007), or counting the number of intervening syntactic heads (Alexopoulou & Keller, 2007). Note that processing difficulty estimates for locality costs depend on the grammar formalism used to describe the linguistic structure.

### *4.3. Short term memory load*

A related idea is that processing may slow down under high memory load – if there are many open dependencies, these may have to be maintained in working memory, and the process of maintaining entities in working memory may overall slow down processing. This idea is formalized as part of the storage cost component of DLT (Gibson, 1998, 2000). Earlier proposals suggested that processing difficulty may be related to the depth of embedding of the current word, and this idea has also been taken up and tested in later proposals, e.g., (Pynte, New, & Kennedy, 2008; Schuler, AbdelRahman, Miller, & Schwartz, 2010), where processing difficulty is hypothesized to occur when a new dependency needs to be added to the memory, or a stored dependent needs to be retrieved from memory for integration. In DLT (see Section 6.1), a combination of integration cost and storage costs is used to also explain how ambiguous input is resolved (preference for structures with lower memory requirements). Storage cost depend on the grammar formalism over which it operates, as different grammar formalisms may suggest flatter vs. deeper structures or may posit a different number of open dependencies (e.g., differences in subcategorization frames affect the number of expected open dependencies) for a given linguistic input.

### *4.4. Frequency-based linking*

Frequencies have early on been found to significantly correlate with processing difficulty and reading times (slower reading on infrequent words). Frequency-based linking hypotheses hold that frequency effects may not only occur for lexical access but also for syntactic structures (more frequent structures are processed faster than infrequent ones). Underlyingly, this could for instance be due to faster memory retrieval for words and structures that are used more often and may thus be easier to activate, or to the retrieval of larger structures, so that less online integration is necessary. Jurafsky (1996) first proposed to use probabilistic context free grammars (PCFGs)4 to estimate probabilities of alternative analyses and used the probabilities to explain garden path sentences: Only the most probable parses (according to the PCFG) would be kept in memory. The improbable ones were pruned using beam search, which discards highly improbable analyses. For interpretations that were pruned, the parser would have to backtrack, which explains the processing difficulty for garden path sentences.

Frequency estimates for linguistic structures obviously very strongly depend on the form and size of linguistic structures that are assumed by the grammar formalism. Furthermore, frequency estimates also depend on the corpus from which they are estimated.

### *4.5. Conflict*

A question that arises in the construction of probability-based models is whether probabilities from different sources (n-grams, parsing, valency) are needed, and how to combine them. Narayanan and Jurafsky (2002) use a belief network with probabilities from a range of different sources. A belief network is more powerful than a PCFG[4] but it is harder to justify the contributing factors and their relationship to one another.

Padó, Crocker, and Keller (2009) suggest a linking hypothesis according to which processing difficulty ensues if probabilities from two different processing components (e.g., syntactic structure and semantics) do not agree with each other, i.e., when they bias the interpretation towards different analyses. This linking hypothesis potentially has some points of overlap with the competition models, except that for the conflict hypothesis, the source of the competition between alternative analyses stems from potentially different processes within comprehension (syntax vs. semantics). This linking hypothesis hence requires an architecture in which syntax and semantics (or whatever domains from which costly conflicts can arise) are different processes.

[4] A PCFG is a context-free grammar in which each grammar rule is associated with a probability. The overall probability of a parse tree is defined as the product of the probabilities of all grammar rules that are applied when generating the tree.

### *4.6. Information content and surprisal*

A measure that has received a lot of attention in the sentence processing literature is surprisal (Hale, 2001; Levy, 2008a). Surprisal quantifies the amount of new information conveyed by a word in context, and the surprisal at word $w_i$ is formalized as $-\log P(w_i|w_1 \ldots w_{i-1})$, i.e., as the probability of observing word $w_i$ given that words $w_1 \ldots w_{i-1}$ have already been processed. The linking hypothesis for surprisal is hence that processing difficulty is proportional to the amount of new information that needs to be processed. Surprisal as such is agnostic with respect to the grammar formalism that is used to as a processing mechanism to derive the predictability estimates, but just like for frequencies, actual surprisal values depend both on the grammar formalism and the data used to estimate the model. Using surprisal as a linking hypothesis also puts constraints on the processing algorithm, which needs to be incremental in order to properly estimate the probability of a word given previous content. An extension of the concept of surprisal is the *noisy channel hypothesis* (Levy, 2008b), which proposes that language comprehenders always maintain some uncertainty about the linguistic input. This overall uncertainty about the input may be due to uncertainty in visual perception during reading, uncertainty about auditory processing due to noise or uncertainty due to own cognitive processes (e.g., imperfect encoding or short term memory). With a noisy channel extension to the model, the identity of word $w_i$ and earlier words that the model conditions on are not certain, but can themselves be described using a probability distribution. However, no broad-coverage noisy channel model exists, as it is unclear to date how to adequately estimate the probabilities of the input observations.

### *4.7. Uncertainty and competing analyses*

Another family of linking hypotheses is related to the idea that ambiguity in the linguistic input may cause processing difficulty: The competition hypothesis suggests that it may be difficult if two or more alternative linguistic analyses have a similar likelihood. These analyses may then compete against one another till one of the alternatives reaches criterion. The same basic concept is also reflected in using *entropy* as a measure for processing difficulty: processing difficulty at a word is suggested to be correlated to the amount of entropy at that word. There are different notions however regarding what entropy is meant: entropy with respect to alternative analyses up to the current word (this is most similar to the competition models), entropy about the upcoming next word (Roark, Bachrach, Cardenas, & Pallier, 2009; Willems, Frank, Nijhof, Hagoort, & Van den Bosch, 2016), or entropy about the rest of the sentence (Linzen & Jaeger, 2016).

### *4.8. Entropy reduction*

The entropy reduction (ER) hypothesis (Hale, 2003, 2006) suggests that processing difficulty may be related to changing from a state of high uncertainty about the rest of the sentence to a state of lower uncertainty about the rest of the sentence during parsing. There are however some conceptual difficulties with this notion (e.g., uncertainty can also increase when processing a sentence, which is not related to any cost or benefit), as discussed by Linzen (2015) and Linzen and Jaeger (2016). Additionally, there is the practical difficulty that entropy with respect to the complete rest of the sentence is extremely difficult to estimate. A number of authors have empirically evaluated the entropy reduction hypothesis (S. L. Frank, 2013; Linzen & Jaeger, 2016; Roark et al., 2009); note however that these implementations are based on different parsers, and use different amounts of look-ahead (entropy reduction with respect to the next word, next four words or whole sentence). Linzen and Jaeger (2016) show that the amount of look-ahead massively affects ER estimates.

[Table 22.1 here]

## 5. Data sets and evaluation

### *5.1. Test sets*

A key challenge that needs to be addressed in order to develop cognitively plausible models of human language processing concerns test data and model evaluation. Here, the state of the art in psycholinguistic modeling lags significantly behind standards in computational linguistics. Most of the models that will be discussed in Section 6 have not been evaluated rigorously. The authors typically describe their performance on a small set of hand-picked examples; often, no attempt is made to test on a range of items from the experimental literature and to evaluate directly against experimentally obtained measures of human language processing (e.g., reading times). This makes it very hard to obtain a realistic estimate of how well the models achieve their aim of capturing human language processing.

More recent work in psycholinguistic modeling has started to use standardized test sets for model evaluation, similar to what is commonplace for tasks in computational linguistics. In computational linguistics, parsers are evaluated against the Penn Treebank, word sense disambiguation systems against the SemEval data sets, co-reference systems against the Tipster or ACE corpora, etc. In psycholinguistic modeling, evaluation is more complicated: two types of test data are required. The first type of test data consists of a collection of representative experimental results. Such datasets should contain the actual experimental materials (sentences or discourse fragments) used in the experiments, together with the experimental measurements obtained (reading times, eye-movement records, ERP data, etc.). The experiments included in this test set should be chosen to cover a wide range of experimental phenomena, e.g., garden paths, syntactic complexity, memory effects (see Section 3). Such test sets enable the standardized evaluation of psycholinguistic models by comparing the model predictions (rankings, surprisal values, memory costs, etc.) against experimental measures on a large set of items. This way both the coverage of a model (how many phenomena can it account for) and its accuracy (how well does it fit the experimental data) can be assessed.

Experimental test sets should be complemented by test sets based on corpus data. In order to assess the efficiency, robustness, and broad coverage of a model, a corpus of unrestricted, naturally occurring text is required. The use of contextualized language data makes it possible to assess not only syntactic models, but also models that capture discourse effects. These corpora need to be annotated with experimental measures, e.g., eye-tracking or reading time data. Some relevant corpora have already been constructed, see the overview in Table 1, and various authors have used them for model evaluation (Demberg & Keller, 2008a; Ferrara Boston, Hale, Kliegl, Patil, & Vasishth, 2008; S. L. Frank, 2009; J. Mitchell, Lapata, Demberg, & Keller, 2010; Patil, Vasishth, & Kliegl, 2009; Pynte et al., 2008; Roark et al., 2009).

### *5.2. Behavioral and neural data*

As outlined in the previous section, a number of authors have evaluated psycholinguistic models against eye-tracking or reading time corpora. Part of the data and evaluation challenge is to extend this evaluation to neural data as provided by event-related potential (ERP) or brain imaging studies (e.g., using functional magnetic resonance imaging, fMRI). Neural data sets are considerably more complex than behavioral ones, and modeling them is an important new task that the community is only beginning to address. Some recent work has evaluated models of word semantics against ERP (S. Frank, Otten, Galli, & Vigliocco, 2015; Murphy, Baroni, & Poesio, 2009) or fMRI or MEG data (Hale, Lutz, Luh, & Brennan, 2015; T. M. Mitchell et al., 2008; van Schijndel, Murphy, & Schuler, 2015; Wehbe, Ashish, Knight, & Mitchell, 2015; Willems et al., 2016). This is a very promising direction, and the challenge is to extend this approach to the sentence and discourse level (see Bachrach, 2008). Again, it will be necessary to develop standardized test sets of both experimental data and corpus data.

### *5.3. Evaluation measures*

We also anticipate that the availability of new test data sets will facilitate the development of new evaluation measures that specifically test the validity of psycholinguistic models. Established evaluation measures from computational linguistics such as Parseval are of limited use, as they can only test the linguistic, but not the behavioral or neural predictions of a model. So far, many authors have relied on qualitative evaluation: if a model predicts a difference in (for instance) reading time between two types of sentences where such a difference was also found experimentally, then that counts as a successful test. In most cases, no quantitative evaluation is performed, as this would require modeling the reading times for individual items and individual participants. Suitable procedures for performing such tests do not currently exist, though linear mixed effects models (Baayen, Davidson, & Bates, 2008) provide a way of dealing with item and participant variation.

Further issues arise from the fact that we often want to compare model fit for multiple experiments (ideally without reparametrizing the models), and that various mutually dependent measures are used for evaluation, e.g., processing effort at the sentence, word, and character level. An important open challenge in this area is to develop evaluation measures and associated statistical procedures that can deal with these problems.

## 6. Models

The earliest approaches were *ranking-based models*, which make psycholinguistic predictions based on the ranking of the syntactic analyses produced by a probabilistic parser. Jurafsky (1996) assumes that processing difficulty is triggered if the correct analysis falls below a certain probability threshold (i.e., is pruned by the parser). Similarly, Crocker and Brants (2000) assume that processing difficulty ensues if the highest-ranked analysis changes from one word to the next. Both approaches have been shown to successfully model garden path effects. Being based on probabilistic parsing techniques, ranking-based models generally achieve a broad coverage, but their efficiency and robustness has not been evaluated. Also, they are not designed to capture syntactic prediction or memory effects (other than search with a narrow beam in Brants & Crocker, 2000).

The ranking-based approach has been generalized by *surprisal models*, which predict processing difficulty based on the change in the probability distribution over possible analyses from one word to the next (Demberg & Keller, 2008a; Ferrara Boston et al., 2008; Hale, 2001; Levy, 2008a; Roark et al., 2009). These models have been successful in accounting for a range of experimental data, and they achieve broad coverage. They also instantiate a limited form of prediction, viz., they build up expectations about the next word in the input. On the other hand, the efficiency and robustness of these models has largely not been evaluated, and memory costs are not modeled (again except for restrictions in beam size).

*Prediction-based models* explicitly predict syntactic structure for upcoming words (Demberg & Keller, 2008b, 2009), thus accounting for experimental results on predictive language processing. This approach implements a strict form of incrementality by building fully connected trees. Memory costs are modeled directly as a distance-based penalty that is incurred when a prediction has to be verified later in the sentence.

The *stack-based model* (Schuler et al., 2010) imposes explicit, cognitively motivated memory constraints on the parser, in effect limiting the stack size available to the parser. This delivers robustness, efficiency, and broad coverage, but does not model syntactic *prediction*. A series of psycholinguistic evaluation studies that build on this model have recently been carried out (van Schijndel, Exley, & Schuler, 2013; van Schijndel et al., 2013; van Schijndel et al., 2015).

### *6.1. Dependency Locality Theory*

Dependency Locality Theory (DLT), suggested by Gibson (1998), operates on top of a dependency analysis. Processing difficulty according to DLT is related to the use of the computational resources consumed by the processor. Two distinct cost components can be distinguished: (i) integration cost associated with encoding new referents and with integrating referents into the structures already built at a given stage in the computation, and (ii) storage cost involved in the storage of parts of the input that may be used in parsing later parts of an input.

Integration costs and storage cost interact through the concepts of energy units, memory units and time units. According to the model, there is only a limited number of energy units available at each point in time, so working memory resources can be used up by having to remember many dependencies (thus using up lots of memory units), in which case there will be less resources for actual integrations (as measured using integration cost), in turn causing them to take more time. The relationship between energy units, memory units and time units was formalized as EU = MU × TU. In the case of ambiguity, analyses that require fewer energy units are preferred.

DLT has been shown to account for a range of linguistic effects including the subject/object relative clause asymmetry, difficulty of center embeddings, garden paths, filler-gap dependencies, heavy NP shift, and extraposition (Gibson, 1998, 2000; Warren & Gibson, 2002; Roark et al., 2009). Broad coverage implementations exist only of the DLT integration cost component (which Gibson, 1998, suggests is the more important component), based on the MiniPar parser (Demberg & Keller, 2008a). A later reimplementation calculates integration cost on top of Stanford dependencies, which achieves better performance (see Demberg, 2013). The broad coverage evaluations showed that there is scant evidence for an effect of integration cost on reading times in general texts (see Demberg & Keller, 2008a, for a detailed analysis, as well as confirmatory evidence from van Schijndel & Schuler, 2013).

### *6.2. Information-theoretic models*

Information-theoretic measures (surprisal, entropy, entropy reduction, see Section 4.6) can be calculated based on incremental generative language models. A first implemented (toy) model, based on a PCFG, was suggested by Hale (2001). As surprisal as an information-theoretic measure is in principle agnostic with respect to the underlying way of estimating information content (i.e., based on word sequences, syntactic structures, semantic expectancies, see also Levy, 2008a), various alternative model implementations have been proposed, including top-down PCFG parsing (Roark, 2001; Roark et al., 2009), dependency-parser based surprisal model

(Ferrara Boston et al., 2008), tree-adjoining grammar based (Demberg et al., 2013), generalized categorial grammars (van Schijndel & Schuler, 2013) as well as sequence model working on POS tag sequences (S. L. Frank & Bod, 2011) (see also chapter 21 by Frank, Monaghan and Tsoukala in this volume, who introduce this and related models in more detail). Most current models include representations of hierarchical syntactic structure for surprisal prediction, as this has been shown to more accurately fit human data (for more detail on a controversy of the empirical usefulness of representing hierarchical structure in information theoretic models, see S. L. Frank & Bod, 2011, and Fossum & Levy, 2012).

Of course there is no specific reason why surprisal estimates should only model lexical information and syntax. Suggestions for how to include semantic information into surprisal models have been proposed (Blacoe & Lapata, 2012; J. Mitchell et al., 2010; Sayeed, Fischer, & Demberg, 2015).

In order to determine surprisal accurately, full parallel search would be necessary. Due to performance reasons, however, all widely used parsers use beam search, i.e., they do not compute all possible analyses to determine surprisal. Some of these models (Roark et al., 2009; S. L. Frank & Bod, 2011) have also been used to calculate related information-theoretic measures such as entropy and entropy reduction. A particular challenge for good estimates of entropy or entropy reduction is that one would need to estimate the entropy of the whole rest of the sentence, i.e., we would need to calculate a probability distribution over the unseen rest of the sentence at each word. As this is very hard and compute-intensive to estimate, most implementations only regard a limited context of next word or next few words, see also Linzen & Jaeger (2016).

Surprisal models have been shown to explain a wide range of psycholinguistic phenomena (see Levy, 2008a, for a more detailed discussion), and furthermore perform well on broad-coverage modeling on eye-tracking corpora (Demberg & Keller, 2008a; Ferrara Boston et al., 2008; Roark et al., 2009; van Schijndel & Schuler, 2013). S. Frank et al. (2015) furthermore showed that surprisal estimates are predictive of the size of the ERP component N400.

### *6.3. Psycholinguistically motivated tree-adjoining grammar*

Psycholinguistically Motivated Tree-adjoining Grammar (PLTAG) is a computational model that explicitly accounts for prediction effects in human parsing. It is psycholinguistically motivated in that it simulates strict incrementality and incorporates an explicit prediction and verification process, as well as memory decay (see Section 1.2 for a discussion of these key properties of human sentence processing). The PLTAG model consists of a grammar formalism, also called PLTAG (Demberg & Keller, 2008b, see Section 2.1.2), together with a parser and a probability model for this formalism (Demberg et al., 2013), and a linking theory that relates PLTAG to behavioral data (Demberg & Keller, 2009).

PLTAG has two mechanisms that account for processing difficulty. First, surprisal is used to quantify difficulty in terms of updates to the parser's probability distribution over possible analyses as the sentence unfolds. Surprisal can be calculated straightforwardly from the probabilities of the (always fully connected) prefix trees of a sentence (a prefix tree is a partial tree that spans a prefix of a sentence). Second, difficulty can arise at verification time, when previously predicted structures are checked against what is actually encountered in the input. By hypothesis, the amount of difficulty generated by verification depends on (a) how difficult the prediction was: the less probable the predicted structure, the higher the verification difficulty;

and (b) how recently the prediction was made: the more the prediction has decayed, the higher the verification difficulty.

In order to model memory decay, the processor needs to keep track of when each syntactic node was predicted. Technically, this is realized by assuming that a predicted node carries a timestamp that is initialized when the node is committed to memory. Based on the timestamp, the duration between initial prediction and verification can be quantified. The verification process thus attributes difficulty to the memory retrieval process that happens when newly encountered evidence is matched against structure whose representation has decayed since it was last activated.

It is instructive to compare PLTAG verification cost to DLT integration cost. While both are distance-based costs, they differ in how distance is calculated. In DLT, distance is a linear function of the number of discourse referents between an argument and its head; verification cost on the other hand is computed using an exponential decay function based on the number of words intervening between the point at which the prediction was made and the point at which it is verified. Furthermore, verification cost not only depends on distance, but also on the difficulty of the prediction. This is modeled in terms of the probability of the predicted structure: the lower the probability of the predicted structure, the higher the cost of verifying it (see Demberg et al., 2013, for a formalization of this intuition).

PLTAG and DLT therefore differ in their assumptions about what causes processing difficulty: DLT integration cost attributes difficulty to integrating the head and its dependents. In PLTAG, the relevant syntactic structure has already been predicted at the point of verification, hence verification cost is not about integrating structure (which we assume to be accounted for by surprisal), but about retrieving previously predicted structure and making sure that only analyses that are consistent with that predicted structure are built.

From the perspective of the comprehension system, the purpose of the verification operation is to check if the predicted input matches the actually observed input, and hence to ensure the consistency of the overall analysis. Verification is an additional operation that a non-predictive parser does not require; it therefore makes sense that verification carries a cost, and that this cost depends on the difficulty of the memory access required. Note that this cost can be safely assumed to be lower than the cost of accidentally constructing inconsistent analyses based on unobserved input, which are then never verified (perhaps until the end of the sentence). It turns out that DLT integration cost and PLTAG verification cost often assign difficulty at the same points in a sentence: in PLTAG, a head is predicted when its dependent (or a dependent of its dependent, as needed to maintaining connectedness) is encountered. This prediction is then verified when the argument is encountered, which is exactly the point at which the integration is assumed to take place in DLT.

PLTAG hence provides a way of combining surprisal-based processing cost with distance-based processing cost. It assumes that the overall processing cost of a word is the sum of the surprisal and the verification cost incurred at that word. Given that surprisal and distance-based cost have been argued to be complementary (Demberg & Keller, 2008a; Staub, 2010), PLTAG can be expected to provide a more complete account than either surprisal or DLT, i.e., it should be able to capture a wider range of experimental results. At the same time, PLTAG provides a unified theoretical framework that explains both types of processing cost: verification cost is a natural consequence of assuming a prediction-based parser, and a prediction-based parser is in turn a consequence of assuming that human sentence processing is incremental and builds fully connected structures. Furthermore, PLTAG allows the two types of cost to be computed over the same representations (tree-adjoining grammar structures): surprisal is associated with the forward-looking, predictive aspect of TAG parsing, while verification is the backward-looking, integrative aspect of TAG parsing. An alternative would be an additive combination of surprisal and DLT (for example); however, such an approach fails to provide a single theoretical framework (it has no way of explaining why the two types of cost arise and how they are related) and requires two separate representations (a phrase structure representation for surprisal and a dependency representation for DLT integration cost).

There is also independent experimental evidence that points towards an integrated model of surprisal-based and distance-based processing cost. For instance, Vasishth and Drenhaus (2011), using a number of experimental paradigms (self-paced reading, eye-tracking, and event-related potentials), investigate relative clause processing in German and find standard distance-based effects at the relative clause verb (as predicted by DLT). However, they also report a surprisal-based effect in first-pass regression probabilities in their eye-tracking study (increased distance between the verb and its argument, i.e., higher predictability, corresponds to fewer regressions), concluding that a complete theory needs to integrate both types of effects. This is corroborated by recent findings by Levy and Keller (2013), who report eye-tracking data for ditransitive sentences in German which indicate not only a decrease in processing difficulty at the verb with an increasing number of verb arguments (as predicted by surprisal), but also an increase in processing difficulty with an increasing distance between the arguments and the verb (as predicted by DLT integration cost). These results strongly suggest that a complete theory needs to combine both types of mechanisms.

### *6.4. Similarity-based interference*

Lewis and Vasishth (2005) propose a model of sentence processing that uses the cognitive architecture ACT-R (Anderson, 1996). ACT-R implements cognitively plausible mechanisms, such as a working memory architecture. ACT-R has also been used to model a large range of other cognitive processes. The *memory and activation model* attempts to explain processing phenomena through memory retrieval effects. The underlying mechanisms of memory retrieval are rehearsal, spreading activation and decay. The model grammar is a PCFG, and it uses left-corner parsing as a processing algorithm to determine top-down predictions about what types of words or structures are needed to build a sentence, simultaneously with bottom-up evidence for what words are encountered in the input. When a word is retrieved from memory, its activation is boosted (this explains e.g., lexical frequency effects: items that are retrieved very often have higher activation). The model also implements a steady activation decay according to the power law of forgetting, which is applied to all of the items in memory.

The model accounts for locality effects (like the subject/object relative clause asymmetry and center embedding) through decay and resulting lower activation of words that need to be retrieved for integration after seeing a lot of intervening material. It can also account for some anti-locality effects through activation of the head through intervening arguments. Furthermore, the theory can explain interference effects (retrieval is hindered by activation of similar items) and storage load effects (if more items need to be stored, there are also more interference effects at retrieval).

### *6.5. Other models*

Another incremental broad-coverage model is the hierarchical sequential model using categorial grammar, also known as the ModelBlocks parser (Wu, Bachrach, Cardenas, & Schuler, 2010; van Schijndel & Schuler, 2013; van Schijndel et al., 2013). A difficulty for incremental parsing is that linguistic structure (in English) is predominantly right-branching. With a right-branching structure, one cannot build a fully connected structure for partial sentences read from the left. The ModelBlocks parser deals with this challenge by performing a right-corner transform, i.e., it rewrites syntax trees such that all right branching structures are turned into left branching structures. Parsing with this model is then a hierarchical sequence process that keeps track of the parsing states and can increase or decrease the size of the stack (the size of the stack reflects the number of open dependencies). Processing difficulty from this model is related to whether stack depth is increased or decreased, with the empirical finding that increasing stack depth leads to increased difficulty and decreased stack depth leads to a facilitation, which is the opposite of the prediction in locality theory (Wu et al., 2010; van Schijndel & Schuler, 2013).

A linking theory has also been proposed for top-down parsing with *minimalist grammars* (Kobele, Gerth, & Hale, 2013): the amount of time (i.e., the number of steps during parsing) that a derivation tree node spends on the parser's stack before being popped is related to increased processing difficulty in the form of higher memory demands.

More details on different incarnations of *competition models* are provided by McRae, Spivey-Knowlton, and Tanenhaus (1998), Tabor, Juliano, and Tanenhaus (1997), and Tabor and Tanenhaus (2001), but see van Gompel, Pickering, Pearson, and Liversedge (2005) who provide an overview of competition models and argue against them. Recently, Lau, Clark, and Lappin (2017) proposed a bottom-up probabilistic parser as a model of gradient grammaticality judgments.

### *6.6. Summary*

To summarize, the challenge is to develop a computational model that captures the key properties of human language processing (see Section 1.2). Table 2 lists these key properties, along with some representative references to experimental work demonstrating these key properties. The table also given a qualitative evaluation of four main model types (ranking, surprisal, prediction, and stack-based models) with respect to the properties.

[Table 22.2 here]

## **7. Discussion**

In this article, we discussed the outlined the state of the art in computational modeling of human sentence processing. Developing computational models is of scientific importance in so far as models are implemented theories: models of language processing allow us to test scientific hypothesis about the cognitive processes that underpin language processing. We argued that this type of precise, formalized hypothesis testing is only possible if standardized data sets and uniform evaluation procedures are available. Ultimately, this approach enables qualitative and quantitative comparisons between theories, and thus enhances our understanding of a key aspect of human cognition, language processing.

In this article, we have focused on syntactic processing. However, there is strong evidence that human language processing is driven by an interaction of syntactic, semantic, and discourse processes. There is considerable experimental evidence that verb senses, selectional restrictions, and thematic roles interact with syntactic ambiguity resolution. Another large body of research has elucidated the interaction of discourse processing and syntactic processing. A key challenge in computational psycholinguistics is therefore the development of language processing models that combine syntactic processing with semantic and discourse processing. So far, this challenge is largely unmet: there are some examples of models that integrate semantic processes such as thematic role assignment into a parsing model (Narayanan & Jurafsky, 2002; Padó et al., 2009; Konstas & Keller, 2015), but models that combine syntactic and discourse are virtually non-existent.

A second challenge for the field is to bring together modeling work in language processing and language development. The human language processor is the product of an acquisition process that does not require explicit training and has access to only limited data: children aged 12–36 months are exposed to between 10 and 35 million words of input (Hart & Risley, 1995). The challenge therefore is to develop a model of language acquisition that works with such small data sets, while also giving rise to a language processor that meets the key criteria in Table 2.

Finally, it remains to note that there is also an applied side to the modeling enterprise. Once computational models of human language processing are available, they can be used to predict the difficulty that humans experience when processing text or speech. This is useful for a number applications: for instance, natural language generation would benefit from being able to assess whether machine-generated text or speech is easy to process. For text simplification (e.g., for children or impaired readers), such a model is even more essential. It could also be used to assess the readability of text, which is of interest in educational applications (e.g., essay scoring). In machine translation, evaluating the fluency of system output is crucial, and a model that predicts processing difficulty could be used for this, or to guide the choice between alternative translations, and maybe even to inform human post-editing.

## References

Abney, S. P., & Johnson, M. (1991). Memory requirements and local ambiguities of parsing strategies. *Journal of Psycholinguistic Research, 20*(3), 233-250.

Alexopoulou, T., & Keller, F. (2007). Locality, cyclicity and resumption: At the interface between the grammar and the human sentence processor. *Language, 83*(1), 110-160.

Altmann, G. T. M., & Steedman, M. J. (1988). Interaction with context during human sentence processing. *Cognition, 30*(3), 191-238.

Anderson, J. R. (1996). Act: A simple theory of complex cognition. *American Psychologist, 51*(4), 355.

Baayen, R. H., Davidson, D. J., & Bates, D. M. (2008). Mixed-effects modeling with crossed random effects for subjects and items. *Journal of Memory and Language, 59*(4), 390-412.

Bachrach, A. (2008). Imaging neural correlates of syntactic complexity in a naturalistic context (Unpublished doctoral dissertation). Massachusetts Institute of Technology, Cambridge, MA.

Barrett, M. J., Agic, Z., & Sogaard, A. (2015). The Dundee treebank. In *Proceedings of the 14th International Workshop on Treebanks and Linguistic Theories* (pp. 242-248). Warsaw, Poland.

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

Blacoe, W., & Lapata, M. (2012). A comparison of vector-based representations for semantic composition. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning* (pp. 546-556). Jeju Island, Korea.

Brants, T., & Crocker, M. W. (2000). Probabilistic parsing and psychological plausibility. In *Proceedings of the 18th International Conference on Computational Linguistics* (pp. 111-117). Saarbrücken/Luxembourg/Nancy.

Chomsky, N. (1957). *Syntactic structures*. The Hague: Mouton.

Crocker, M. W., & Brants, T. (2000). Wide-coverage probabilistic sentence processing. *Journal of Psycholinguistic Research, 29*(6), 647-669.

Delong, K. A., Urbach, T. P., & Kutas, M. (2005). Probabilistic word pre-activation during language comprehension inferred from electrical brain activity. *Nature Neuroscience, 8*, 1117-1121.

Demberg, V. (2013). Integration Costs on Auxiliaries? - a self-paced reading study using WebExp. In M. Knauff, M. Pauen, N. Sebanz, & I. Wachsmuth (Eds.), *Proceedings of the 35th annual conference of the cognitive science society* (pp. 2160-2165). Berlin, Germany: Austin TX: Cognitive Science Society.

Demberg, V., & Keller, F. (2008a). Data from eye-tracking corpora as evidence for theories of syntactic processing complexity. *Cognition, 101*(2), 193-210.

Demberg, V., & Keller, F. (2008b). A psycholinguistically motivated version of TAG. In *Proceedings of the 9th International Workshop on Tree Adjoining Grammars and Related Formalisms* (pp. 25-32). Tübingen, Germany.

Demberg, V., & Keller, F. (2009). A computational model of prediction in human parsing: Unifying locality and surprisal effects. In N. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st Annual Conference of the Cognitive Science Society* (pp. 1888-1893). Amsterdam, The Netherlands.

Demberg, V., Keller, F., & Koller, A. (2013). Incremental, predictive parsing with psycholinguistically motivated tree-adjoining grammar. *Computational Lingusitics, 39*(4), 1025-1066.

Duffy, S. A., Morris, R. K., & Rayner, K. (1988). Lexical ambiguity and fixation times in reading. *Journal of Memory and Language, 27*, 429-446.

Eady, J., & Fodor, J. (1981, December). Is center embedding a source of processing difficulty. In *Linguistics Society of America Annual Meeting*.

Federmeier, K. D., Wlotko, E. W., De Ochoa-Dewald, E., & Kutas, M. (2007). Multiple effects of sentential constraint on word processing. *Brain research, 1146*, 75-84.

Ferrara Boston, M., Hale, J., Kliegl, R., Patil, U., & Vasishth, S. (2008). Parsing costs as predictors of reading difficulty: An evaluation using the Potsdam Sentence Corpus. *Journal of Eye Movement Research, 2*(1), 1-12.

Ferreira, F., Christianson, K., & Hollingworth, A. (2001). Misinterpretations of garden-path sentences: Implications for models of sentence processing and reanalysis. *Journal of Psycholinguistic Research, 30*(1), 3-20.

Ferreira, F., & Henderson, J. (1991). Recovery from misanalyses of garden-path sentences. *Journal of Memory and Language, 30*(6), 725-745.

Fodor, J. D., & Frazier, L. (1978). The sausage machine: A new two-stage parsing model. *Cognition, 6*, 291-325.

Ford, M., Bresnan, J., & Kaplan, R. M. (1982). A competence-based theory of syntactic closure. In J. Bresnan (Ed.), *The mental representation of grammatical relations* (pp. 727-796). Cambridge, MA: MIT Press.

Fossum, V., & Levy, R. (2012). Sequential vs. hierarchical syntactic models of human incremental sentence processing. In *Proceedings of the 3rd Workshop on Cognitive Modeling and Computational Linguistics* (pp. 61-69). Montreal, Canada.

Frank, S., Otten, L., Galli, G., & Vigliocco, G. (2015). The ERP response to the amount of information conveyed by words in sentences. *Brain and Language, 40*, 1-11.

Frank, S. L. (2009). Surprisal-based comparison between a symbolic and a connectionist model of sentence processing. In N. Taatgen & H. van Rijn (Eds.), *Proceedings of the 31st annual conference of the cognitive science society* (pp. 1139-1144). Amsterdam: Cognitive Science Society.

Frank, S. L. (2013). Uncertainty reduction as a measure of cognitive load in sentence comprehension. *Topics in Cognitive Science, 5*, 475-494.

Frank, S. L., & Bod, R. (2011). Insensitivity of the human sentence-processing system to hierarchical structure. *Psychological science, 22*(6), 829-834.

Frank, S. L., Monsalve, I., Thompson, R., & Vigliocco, G. (2013). Reading-time data for evaluating broad-coverage models of English sentence processing. *Behavior Research Methods, 45*, 1182-1190.

Frazier, L. (1979). *On comprehending sentences: Syntactic parsing strategies* (Unpublished doctoral dissertation). University of Connecticut.

Gibson, E. (1991). *A computational theory of human linguistic processing: Memory limitations and processing breakdown* (Unpublished doctoral dissertation). Carnegie Mellon University, Pittsburgh.

Gibson, E. (1998). Linguistic complexity: locality of syntactic dependencies. *Cognition, 68*, 1-76.

Gibson, E. (2000). Dependency locality theory: A distance-based theory of linguistic complexity. In A. Marantz, Y. Miyashita, & W. O'Neil (Eds.), *Image, language, brain: Papers from the first mind articulation project symposium* (pp. 95-126). Cambridge, MA: MIT Press.

Hale, J. (2001). A probabilistic Earley parser as a psycholinguistic model. In *Proceedings of the 2nd Conference of the North American Chapter of the Association for Computational Linguistics* (Vol. 2, pp. 159-166). Pittsburgh, PA: Association for Computational Linguistics.

Hale, J. (2003). The information conveyed by words. *Journal of Psycholinguistic Research, 32*, 101-122.

Hale, J. (2006). Uncertainty about the rest of the sentence. *Cognitive Science, 30*(4), 609-642.

Hale, J., Lutz, D. E., Luh, W.-M., & Brennan, J. R. (2015). Modeling fMRI time courses with linguistic structure at various grain sizes. In *Proceedings of the 6th Workshop on Cognitive Modeling and Computational Linguistics* (pp. 89-97). Montreal, Canada.

Hart, B., & Risley, T. R. (1995). *Meaningful differences in the everyday experience of young American children*. Baltimore, MD: Paul H. Brookes.

Hobbs, J. R., & Bear, J. (1990). Two principles of parse preference. In *Proceedings of the 13th Conference on Computational linguistics* (pp. 162-167). Helsinki, Finland.

Joshi, A., Levy, L., & Takahashi, M. (1975). Tree adjunct grammars. *Journal of the Computer and System Sciences, 10*(1), 136-163.

Joshi, A., & Schabes, Y. (1997). Tree-adjoining grammars. In G. Rozenberg & A. Salomaa (Eds.), *Handbook of formal languages* (Vol. 3., pp. 69-123). Berlin: Springer.

Jurafsky, D. (1996). A probabilistic model of lexical and syntactic access and disambiguation. *Cognitive Science, 20*(2), 137-194.

Jurafsky, D., & Martin, J. H. (2009). *Speech and language processing: An introduction to natural language processing, computational linguistics and speech recognition (2nd edition)*. Upper Saddle River, NJ: Pearson Education.

Kamide, Y., Altmann, G. T. M., & Haywood, S. L. (2003). The time-course of prediction in incremental sentence processing: Evidence from anticipatory eye movements. *Journal of Memory and Language, 49*, 133-156.

Kamide, Y., Scheepers, C., & Altmann, G. T. M. (2003). Integration of syntactic and semantic information in predictive processing: Cross-linguistic evidence from German and English. *Journal of Psycholinguistic Research, 32*, 37-55.

Kennedy, A., & Pynte, J. (2005). Parafoveal-on-foveal effects in normal reading. *Vision Research, 45*, 153-168.

Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition, 2*(1), 15-47.

Kliegl, R., Nuthmann, A., & Engbert, R. (2006). Tracking the mind during reading: The influence of past, present, and future words on fixation durations. *Journal of Experimental Psychology: General, 135*(1), 12-35.

Kobele, G. M., Gerth, S., & Hale, J. (2013). Memory resource allocation in top-down minimalist parsing. In *Proceedings of the 18th International Conference on Formal Grammar 2013* (pp. 32-51). Springer, Berlin.

Konieczny, L. (2000). Locality and parsing complexity. *Journal of Psycholinguistic Research, 29*(6), 627- 645.

Konieczny, L., & Döring, P. (2003). Anticipation of clause-final heads: Evidence from eye-tracking and SRNs. In P. P. Slezak (Ed.), *Proceedings of the Joint International Conference on Cognitive Science, ICCS/ASCS, Sydney* (pp. 13-17).

Konstas, I., & Keller, F. (2015). Semantic role labeling improves incremental parsing. In *Proceedings of the 53rd annual meeting of the association for computational linguistics* (pp. 1191-1201). Beijing, China.

Lau, J. H., Clark, A., & Lappin, S. (2017). Grammaticality, acceptability, and probability: a probabilistic view of linguistic knowledge. *Cognitive Science, 41*(5), 1202-1241.

Levy, R. (2008a). Expectation-based syntactic comprehension. *Cognition, 106*(3), 1126-1177.

Levy, R. (2008b). A noisy-channel model of rational human sentence comprehension under uncertain input. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 234-243). Honolulu, Hawaii.

Levy, R., Bicknell, K., Slattery, T., & Rayner, K. (2009). Eye movement evidence that readers maintain and act on uncertainty about past linguistic input. *Proceedings of the National Academy of Sciences*, 106(50), 21086-21090.

Levy, R., & Keller, F. (2013). Expectation and locality effects in German verb-final structures. *Journal of Memory and Language, 68*(2), 199-222.

Lewis, R. L. (1996). A theory of grammatical but unacceptable embeddings. *Journal of Psycholinguistic Research, 25*(93), 116.

Lewis, R. L., & Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science, 29*, 1-45.

Linzen, T. (2015). *Probabilistic linguistic representations: Between learning and processing* (Unpublished doctoral dissertation). New York University.

Linzen, T., & Jaeger, T. F. (2016). Uncertainty and expectation in sentence processing: evidence from subcategorization distributions. *Cognitive science, 40*(6), 1382-1411.

MacDonald, M. C., Pearlmutter, N. J., & Seidenberg, M. S. (1994). Lexical nature of syntactic ambiguity resolution. *Psychological Review, 101*, 676-703.

McDonald, R., Pereira, F., Ribarov, K., & Hajič, J. (2005). Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the human language technology conference and the conference on empirical methods in natural language processing* (pp. 523-530). Vancouver, Canada.

McRae, K., Spivey-Knowlton, M. J., & Tanenhaus, M. K. (1998). Modeling the influence of thematic fit (and other constraints) in on-line sentence comprehension. *Journal of Memory and Language, 38*(3), 283-312.

Miller, G., & Isard, S. (1964). Free recall of self-embedded English sentences. *Information and Control, 7*(3), 292-303.

Mitchell, J., Lapata, M., Demberg, V., & Keller, F. (2010). Syntactic and semantic factors in processing difficulty: An integrated measure. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics* (pp. 196-206). Uppsala, Sweden.

Mitchell, T. M., Shinkareva, S. V., Carlson, A., Chang, K.-M., Malave, V. L., Mason, R. A., & Just, M. A. (2008). Predicting human brain activity associated with the meanings of nouns. *Science, 320*(5880), 1191-1195.

Murphy, B., Baroni, M., & Poesio, M. (2009). EEG responds to conceptual stimuli and corpus semantics. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 619-627). Suntec City, Singapore.

Narayanan, S., & Jurafsky, D. (2002). A Bayesian model predicts human parse preference and reading time in sentence processing. In T. G. Dietterich, S. Becker, & Z. Ghahramani (Eds.), *Advances in neural information processing systems 14* (pp. 59-65). Cambridge, MA: MIT Press.

Nivre, J. (2003). An efficient algorithm for projective dependency parsing. In *Proceedings of the international workshop on parsing technologies* (pp. 149-160). Nancy, France.

Padó, U., Crocker, M. W., & Keller, F. (2009). A probabilistic model of semantic plausibility in sentence processing. *Cognitive Science, 33*(5), 794-838.

Patil, U., Vasishth, S., & Kliegl, R. (2009). Compound effect of probabilistic disambiguation and memory retrievals on sentence processing: Evidence from an eye-tracking corpus. In A. Howes, D. Peebles, & R. Cooper (Eds.), *Proceedings of 9th International Conference on Cognitive Modeling*. Manchester, England.

Pynte, J., New, B., & Kennedy, A. (2008). On-line contextual influences during reading normal text: A multiple-regression analysis. *Vision Research, 48*(21), 2172-2183.

Roark, B. (2001). Probabilistic top-down parsing and language modeling. *Computational Linguistics, 27*(2), 249-276.

Roark, B., Bachrach, A., Cardenas, C., & Pallier, C. (2009). Deriving lexical and syntactic expectation-based measures for psycholinguistic modeling via incremental top-down parsing. In *Proceedings of the conference on empirical methods in natural language processing* (pp. 324-333). Suntec City, Singapore.

Sanford, A. J., & Sturt, P. (2002). Depth of processing in language comprehension: Not noticing the evidence. *Trends in Cognitive Sciences, 6*, 382-386.

Sayeed, A., Fischer, S., & Demberg, V. (2015). Vector-space calculation of semantic surprisal for predicting word pronunciation duration. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing* (Vol. 1, pp. 763-773). Beijing, China.

Schuler, W., AbdelRahman, S., Miller, T., & Schwartz, L. (2010). Broad-coverage parsing using human-like memory constraints. *Computational Linguistics, 26*(1), 1-30.

Spivey-Knowlton, M., & Sedivy, J. (1995). Resolving attachment ambiguities with multiple constraints. *Cognition, 55*, 227-267.

Stabler, E. P. (1994). The finite connectivity of linguistic structure. In C. Clifton, L. Frazier, & K. Rayner (Eds.), *Perspectives on Sentence Processing* (pp. 303-336). Hillsdale, NJ: Erlbaum.

Staub, A. (2010). Eye movements and processing difficulty in object relative clauses. *Cognition, 116*, 71-86.

Staub, A., & Clifton, C. (2006). Syntactic prediction in language comprehension: Evidence from either . . . or. *Journal of Experimental Psychology: Learning, Memory, and Cognition, 32*, 425-436.

Sturt, P., & Lombardo, V. (2005). Processing coordinated structures: Incrementality and connectedness. *Cognitive Science, 29*(2), 291-305.

Tabor, W., Galantucci, B., & Richardson, D. (2004). Effects of merely local syntactic coherence on sentence processing. *Journal of Memory and Language, 50*(4), 355-370.

Tabor, W., & Hutchins, S. (2004). Evidence for self-organized sentence processing: Digging in effects. *Journal of Experimental Psychology-Learning Memory and Cognition, 30*(2), 431-450.

Tabor, W., Juliano, C., & Tanenhaus, M. K. (1997). Parsing in a dynamical system: An attractor-based account of the interaction of lexical and structural constraints in sentence processing. *Language and Cognitive Processes, 12*(2-3), 211-271.

Tabor, W., & Tanenhaus, M. K. (2001). Dynamical systems for sentence processing. In M. Christiansen & N. Chater (Eds), *Connectionist psycholinguistics* (pp. 177-211). Westport, CT: Ablex Publishing.

Tanenhaus, M. K., Spivey-Knowlton, M. J., Eberhard, K. M., & Sedivy, J. C. (1995). Integration of visual and linguistic information in spoken language comprehension. *Science, 268*, 1632-1634.

Temperley, D. (2007). Minimization of dependency length in written English. *Cognition, 105*(2), 300-333.

Traxler, M. J., Pickering, M. J., & Clifton, C. (1998). Adjunct attachment is not a form of lexical ambiguity resolution. *Journal of Memory and Language, 39*, 558-592.

van Gompel, R. P., Pickering, M. J., Pearson, J., & Liversedge, S. P. (2005). Evidence against competition during syntactic ambiguity resolution. *Journal of Memory and Language, 52*, 284-307.

van Schijndel, M., Exley, A., & Schuler, W. (2013). A model of language processing as hierarchic sequential prediction. *Topics in Cognitive Science, 5*, 522-540.

van Schijndel, M., & Schuler, W. (2013). An analysis of frequency- and memory-based processing costs. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* (pp. 95-105). Association for Computational Linguistics.

van Schijndel, M., Murphy, B., & Schuler, W. (2015). Evidence of syntactic working memory usage in MEG data. *Proceedings of the 6th Workshop on Cognitive Modeling and Computational Linguistics,* 79-88. Montreal, Canada.

Vasishth, S., & Drenhaus, H. (2011). Locality in German. *Dialogue and Discourse, 1*, 59-82.

Vasishth, S., & Lewis, R. L. (2006). Argument-head distance and processing complexity: Explaining both locality and antilocality effects. *Language, 82*(4), 767-794.

Warren, T., & Gibson, E. (2002). The influence of referential processing on sentence complexity. *Cognition, 85*(1), 79-112.

Wehbe, L., Ashish, V., Knight, K., & Mitchell, T. (2015). Aligning context-based statistical models of language with brain activity during reading. In *Proceedings of the conference on empirical methods in natural language processing*. Lisbon, Portugal.

Willems, R., Frank, S., Nijhof, A., Hagoort, P., & Van den Bosch, A. (2016). Prediction during natural language comprehension. *Cerebral Cortex*, *26*(6), 2506-2516.

Wu, S., Bachrach, A., Cardenas, C., & Schuler, W. (2010). Complexity metrics in an incremental right-corner parser. In *Proceedings of the 48th annual meeting of the association for computational linguistics* (pp. 1189-1198). Uppsala, Sweden.

**Tables**

Table 22.1

Test corpora that have been used for psycholinguistic modeling of sentence processing. Note that the  Potsdam Corpus and the UCL Corpus consist of isolated sentences, rather than of continuous text.  Words: number of word tokens in the corpus; Part: number of participants; PoS: parts of speech; dep: dependency parses.

| **Corpus** | **Language** | **Words** | **Part** | **Method** | **Annotation** | **Reference** |
|---|---|---|---|---|---|---|
| Dundee Corpus | English, French | 50,000 | 10 | Eye-tracking | PoS, dep | Kennedy and Pynte (2005), Barrett et al. (2015) |
| Potsdam Corpus | German | 1,138 | 222 | Eye-tracking | - | Kliegl et al. (2006) |
| MIT Corpus | English | 3,534 | 23 | Self-paced reading | - | Bachrach (2008) |
| UCL Corpus | English | 4,946 | 117 | Self-paced reading | PoS | S. L. Frank et al. (2013) |
| | | 1,931 | 43 | Eye-tracking, ERP | | |

Table 22.2

Key properties of human language processing and their instantiation in various models of sentence processing (see Section 6 for details). Rank: ranking-based models; Surp: surprisal-based models; Pred: prediction-based models; Stack: stack-based models.

| Property | **Evidence** | Model | | | |
|---|---|---|---|---|---|
| | | **Rank** | **Surp** | **Pred** | **Stack** |
| Efficiency and robustness | Ferreira et al. (2001); Sanford and Sturt (2002) | − | − | − | + |
| **Broad coverage** | Crocker and Brants (2000) | + | + | − | + |
| Incrementality and connectedness | Tanenhaus et al. (1995); Konieczny (2000); Sturt and Lombardo (2005) | + | + | + | + |
| Prediction | Kamide, Altmann, and Haywood (2003); Delong et al. (2005); Staub and Clifton (2006) | − | ± | + | − |
| Memory cost | Gibson (1998); Vasishth and Lewis (2006) | − | − | + | + |

**Figure captions**

Figure 22.1

Example for a syntax tree.

Figure 22.2

Example of local ambiguity.

Figure 22.3

Subfigures (a)–(d) show an example for a PLTAG grammar, containing initial and auxiliary trees (shared with standard tree-adjoining grammar) and an example for a prediction tree (which is specific to PLTAG). Subfigure (e) shows an incremental derivation in PLTAG using the trees (a)–(d). Figures taken from Demberg, Keller, and Koller (2013).

Figure 22.4

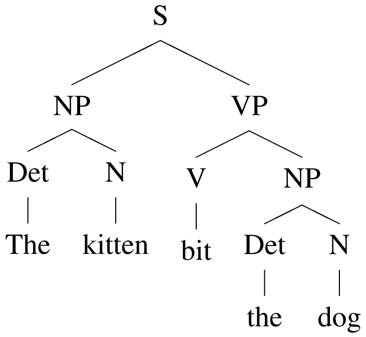
An example for a dependency graph. The arc labels are grammatical relations (subject, object, modifier, etc.), the targets of the arrows are part-of-speech labels (JJ for adjective, NN for noun, VBD for verb, etc.).

Figure 22.5

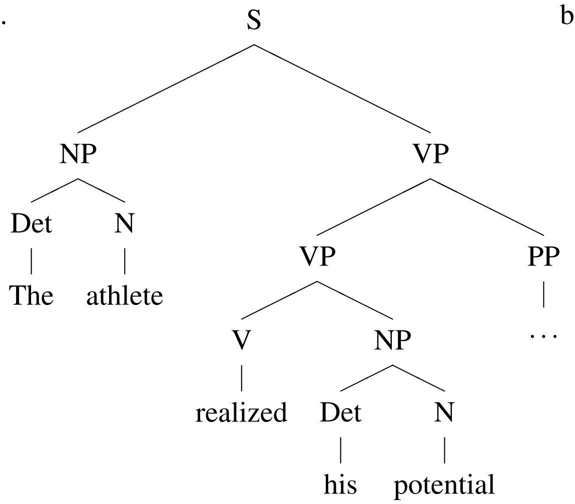
An example for (a) projective and (b) non-projective dependency trees.

Figure 22.6

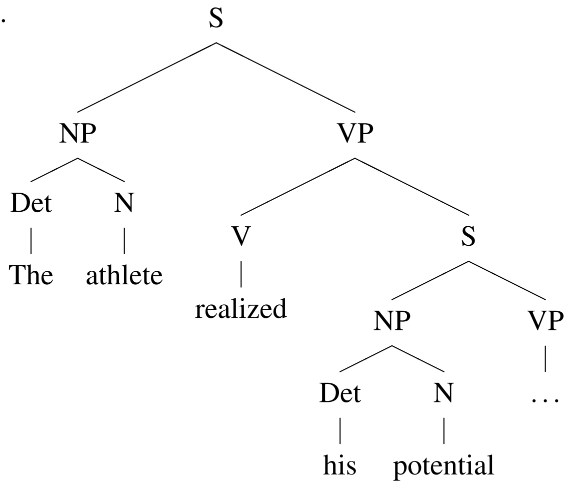
Example for a left-corner derivation.

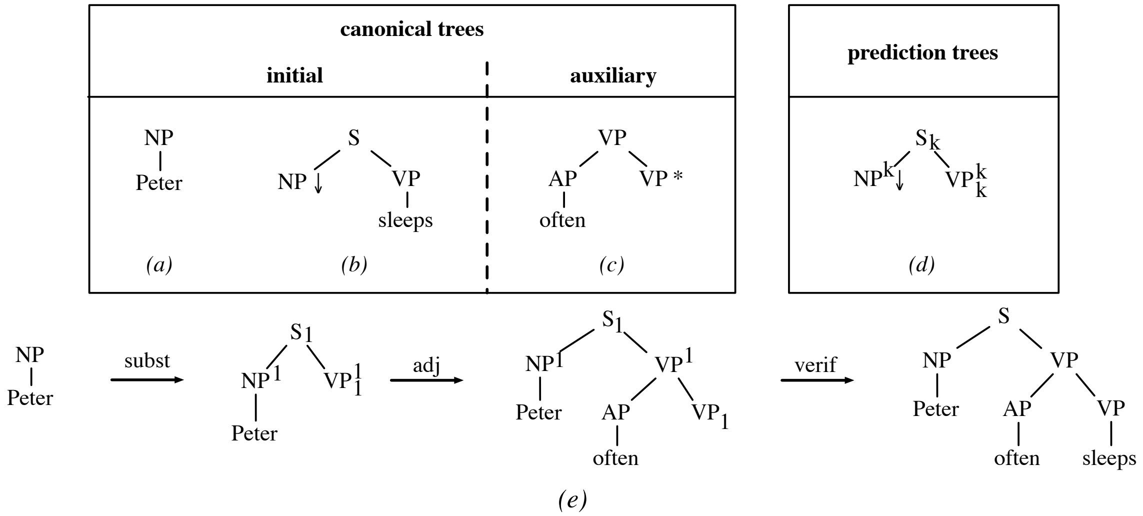

S
NP
VP
Det
N
V
NP
The
kitten
bit
Det
N
the
dog

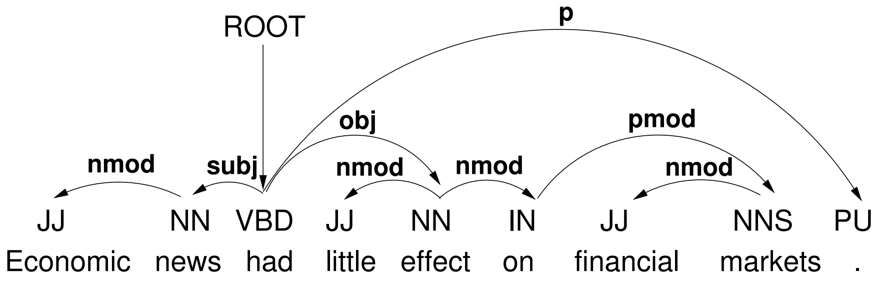
a.

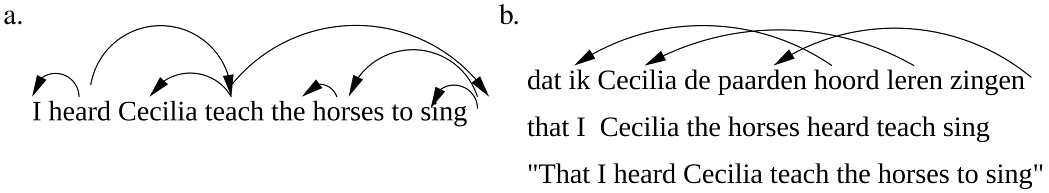

S
NP
VP
Det
N
The
athlete
VP
PP
. . .
V
NP
realized
Det
N
his
potential

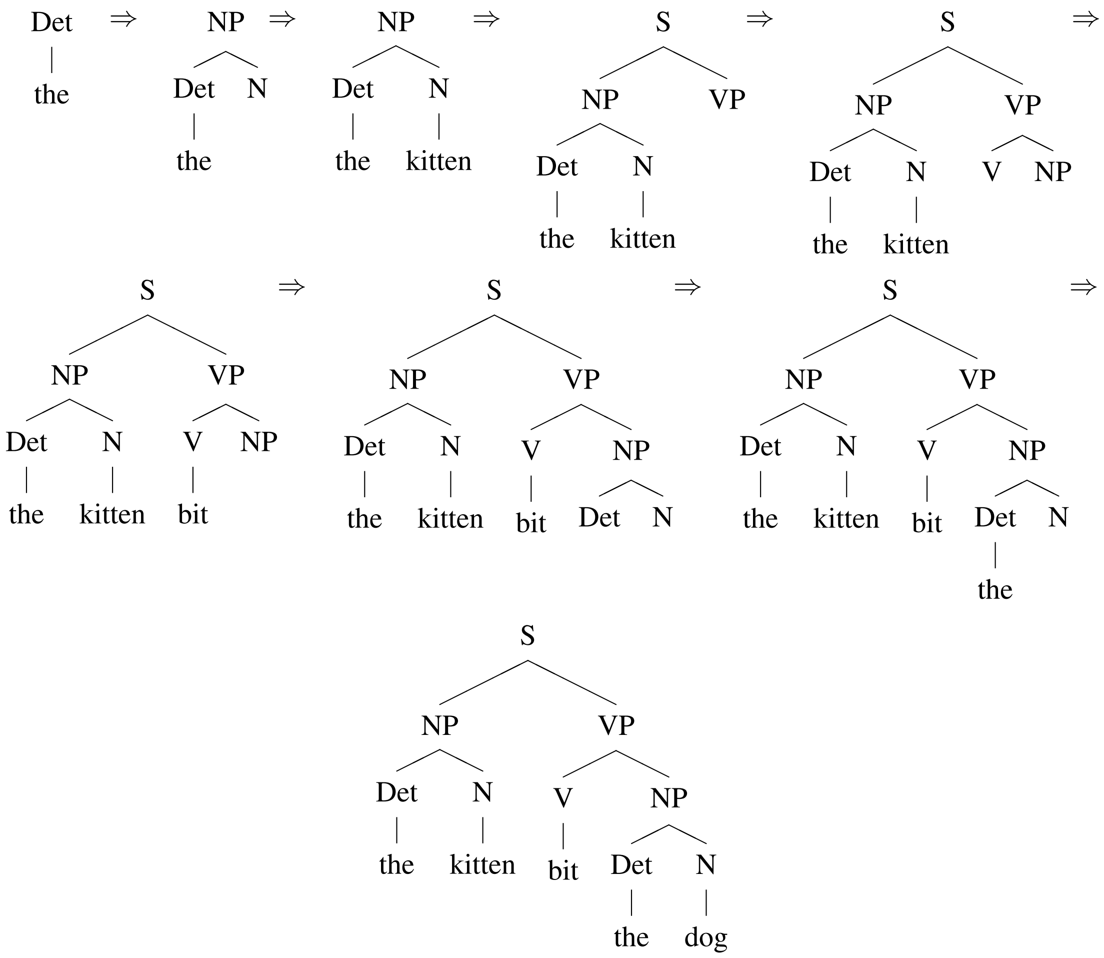
b.


S
NP
VP
Det
N
The
athlete
V
S
realized
NP
VP
. . .
Det
N
his
potential


canonical trees
initial
auxiliary
prediction trees
NP
Peter
S
NP ↓
VP
sleeps
VP
AP
VP *
often
S_k
NP^k ↓
VP^k_k
(a)
(b)
(c)
(d)
NP
Peter
subst
S_1
NP^1
VP^1_1
Peter
adj
S_1
NP^1
VP^1
Peter
AP
VP_1
often
verif
S
NP
VP
Peter
AP
VP
often
sleeps
(e)


ROOT
p
obj
pmod
nmod
subj
nmod
nmod
nmod
JJ
NN
VBD
JJ
NN
IN
JJ
NNS
PU
Economic
news
had
little
effect
on
financial
markets
.


a.
I heard Cecilia teach the horses to sing
b.
dat ik Cecilia de paarden hoord leren zingen
that I  Cecilia the horses heard teach sing
"That I heard Cecilia teach the horses to sing"



Det the ⇒
NP Det N the ⇒
NP Det N the kitten ⇒
S NP VP Det N the kitten ⇒
S NP VP Det N V NP the kitten ⇒
S NP VP Det N V NP the kitten bit ⇒
S NP VP Det N V NP the kitten bit Det N ⇒
S NP VP Det N V NP the kitten bit Det N the ⇒
S NP VP Det N V NP the kitten bit Det N the dog