Word-Level Measures of Syntactic Complexity in Sentence Processing: An Organizing Review

Cas W. Coopmans[1*], Noa Talker-Aharoni[2], Galit Agmon[2,3*]

[1]New York University, New York, USA

[2]Department of English Literature and Linguistics, Bar-Ilan University, Ramat Gan, Israel

[3]The Gonda Center for Multidisciplinary Brain Research, Bar-Ilan University, Ramat Gan, Israel

Abstract

The recent adoption of naturalistic paradigms in the psychology and neuroscience of language has necessitated methods for quantifying incremental syntactic processing complexity. However, there is little consensus about how complexity should be quantified. Existing studies employ a wide range of complexity metrics, reflecting divergent assumptions about the nature of grammatical representations and the mechanisms underlying syntactic processing. Because these assumptions are rarely made explicit, findings across studies cannot be reliably interpreted or consistently synthesized into general conclusions. The goal of the present study is to review the word-level syntactic complexity metrics that have been used to model behavioral and neural measures of syntactic processing. To organize the empirical literature, we introduce a taxonomy based on two complementary dimensions: the grammatical distinction between constituency and dependency representations, and the cognitive distinction between computation- and memory-related aspects of syntactic processing. Our review reveals substantial heterogeneity in how syntactic complexity is operationalized and interpreted, even among ostensibly similar metrics. To test this quantitatively, we applied hierarchical clustering to 26 metrics computed on a sample of 100 annotated sentences. Computation- and memory-related metrics largely clustered together, providing quantitative support for the relevance of our proposed taxonomy. However, this pattern emerged only when constituency- and dependency-based metrics were analyzed separately, showing that the choice of the grammatical formalism shapes how syntactic complexity is quantified. Taken together, these findings underscore the need for greater conceptual clarity in the use of syntactic complexity metrics and for more explicit linking hypotheses connecting linguistic structure to cognitive and neural processes. This review provides a useful framework for achieving these goals.

Keywords: syntax, constituency grammar, dependency grammar, node count, complexity metrics, neurocomputational model

***Correspondence:**
Cas W. Coopmans: cc9248@nyu.edu
Galit Agmon: galit.agmon@biu.ac.il

1. Introduction

A central goal of the neuroscience of language is to explain how linguistic structures are incrementally computed in the brain. Progress on this question is essential not only for understanding language as a neurocognitive faculty, but also for using neural data to adjudicate between competing linguistic theories. To achieve this goal, what is needed are explicit linking hypotheses that relate formally defined properties of linguistic structures and operations to measurable variation in cognitive and neural indices of language processing effort. In particular, such hypotheses must specify how syntactic structure is incrementally constructed, and how the cognitive components of this process continuously vary in time.

Traditionally, psycho- and neurolinguistic studies of syntax have relied on contrast-based designs that compare minimally different sentence types, such as subject- and object-relative clauses. In these paradigms, it is assumed that one construction is structurally more complex than the other and therefore poses greater processing demands. The increase in processing difficulty has been demonstrated across a range of behavioral and neural measures, including longer reading times, lower comprehension accuracy, greater production difficulty, and increased neural activation (e.g., Caplan et al., 2001; Gordon et al., 2001; Grodner & Gibson, 2005; King & Just, 1991; Pallier et al., 2011; Scontras et al., 2014). The experimental literature based on contrast-based paradigms has motivated theories of the syntactic mechanisms involved in language comprehension and production (for reviews, see Bock, 1982; Ferreira & Engelhardt, 2006; Frazier, 1987; Grodzinsky & Friederici, 2006; Hagoort & Indefrey, 2014; Hagoort et al., 1999; Matchin & Hickok, 2020; Townsend & Bever, 2001; Zaccarella et al., 2017). However, because these paradigms often operationalize complexity at the level of entire constructions, they provide only coarse-grained estimates of processing difficulty. As a result, they do not readily distinguish which syntactic operations drive the observed effects, nor do they capture when, during incremental processing, processing costs arise.

Language comprehension proceeds incrementally: syntactic structure and semantic interpretation are continuously updated as each word is encountered. Each incoming word triggers syntactic operations that modify an evolving representation of sentence structure. It is therefore possible to define, for every single word, a measure of syntactic complexity that quantifies the processing demands associated with that word. This measure can then be used as a continuous

regressor of behavioral and neural measures of language processing, including reading times, eye movements, and neuroimaging measures (e.g., fMRI, M/EEG; Brennan, 2016; Hale et al., 2022). Compared to traditional contrast-based designs, these continuous approaches have broad coverage: the complexity metrics are not constrained to a restricted set of constructions but can be defined for every word in any sentence, thus providing corpus-wide estimates of processing demands. This allows for the investigation of language processing in more naturalistic contexts, increasing the ecological validity of established findings (Hamilton & Huth, 2020; Nastase et al., 2020).

Despite the growing use of word-level complexity metrics in psycho- and neurolinguistics, there appears to be little consensus on how syntactic complexity should be operationalized. Existing studies differ substantially both in the grammatical frameworks they adopt and in the specific metrics they derive from them, often either introducing a novel metric or modifying an existing one. Most of the time, this is done without making explicit the underlying theoretical commitments, and without comparing the cognitive validity of the newly introduced metrics against existing alternatives—oftentimes, studies simply claim to study "syntactic processing". Moreover, the few cases that do make their motivations explicit illustrate fundamental disagreement about the cognitive status of grammatical representations. For example, some studies directly compare complexity metrics derived from constituency- and dependency-based representations, thus treating these formalisms as encoding complementary aspects of syntactic structure (Cai et al., 2026; Lopopolo et al., 2021). In contrast, other work treats constituency and dependency structures as isomorphic representations of the same underlying relations (Liu et al., 2026; see Nedft & Baggio, 2024 for discussion). These divergent assumptions reflect fundamentally different views about what aspects of syntactic structure are cognitively represented. Such inconsistency not only limits the comparability of findings across studies, but also hinders progress in linking linguistics and neuroscience (for discussion, see Coopmans & Zaccarella, 2023; Embick & Poeppel, 2015; Nefdt & Baggio, 2024; Poeppel & Embick, 2005; van der Burght et al., 2023).

This paper aims to provide a principled organization of word-level syntactic complexity metrics by clarifying their cognitive assumptions and theoretical commitments. We present a taxonomy of existing metrics that is grammatically grounded in the distinction between constituency and dependency representations, and cognitively grounded in the distinction between

computation and memory (i.e., incremental structure building versus and working memory demands). Building on this framework, we synthesize findings from the neurobiology of language in order to identify neural correlates captured by different metrics, and to describe how these relate to cognitive aspects of syntactic processing. In the spirit of Marr (1982), we believe that an account of the neural basis of syntax ought to be explicit at the computational, algorithmic, and implementational levels of description. Subsequent sections will therefore discuss each of these levels separately, focusing on constituency and dependency grammars as computational-level descriptions of syntactic structure (Section 2), syntactic complexity metrics as algorithmic linking hypotheses that operationalize cognitive processes associated with these representations (Sections 3 and 4), and neural correlates of syntactic processing as implementational-level data (Section 5). The hierarchical clustering analysis in Section 4 reveals that memory and computation are dissociable cognitive components, but only when evaluated separately for constituency- and dependency-based metrics. We conclude by discussing the implications of this framework for the interpretation of naturalistic studies of language processing, and by outlining relevant directions for future research (Section 6).

2. Two models of grammar: Constituency and dependency

Theories of grammar are commonly organized around either the principle of *constituency* or the principle of *dependency*. The resulting grammars—constituency grammars versus dependency grammars—differ both in the kinds of units they treat as fundamental and in the type of relations that connect these units. Constituency grammars take phrases (or constituents) as basic units. Phrases carry a categorial label (e.g., noun phrase, verb phrase) and are related to each other via hierarchical containment relations: smaller constituents are combined into, and thus contained in, larger constituents (Carnie, 2021; Chomsky, 1957; Sag et al., 2003; Steedman, 2000). This type of representation is illustrated in Figure 1, where the sentence *The current user must be an administrator for the computer* is decomposed into constituents that are hierarchically arranged in a tree. Dependency grammars, in contrast, take words rather than phrases as the fundamental unit of structure and represent grammatical relations via word-based dependencies (Gibson, 2025; Hudson, 1990; Mel'čuk, 1988; Tesnière, 1959/2015). In a dependency representation, words are connected via binary, asymmetric relations that link a head to its dependents. Figure 1 shows such

a representation for the same example sentence, in which the asymmetry of dependency relations is expressed via the direction of the arc from head to dependent.

As Figure 1 shows, a critical difference between these representations is that constituency structures contain intermediate, non-terminal nodes (i.e., constituents) while dependency structures do not. In constituency representations, every word corresponds to more than one node, creating hierarchical tree structures with multiple layers of constituency. Dependency structures, in contrast, do not explicitly represent constituents, so each word corresponds to exactly one node in the structure.

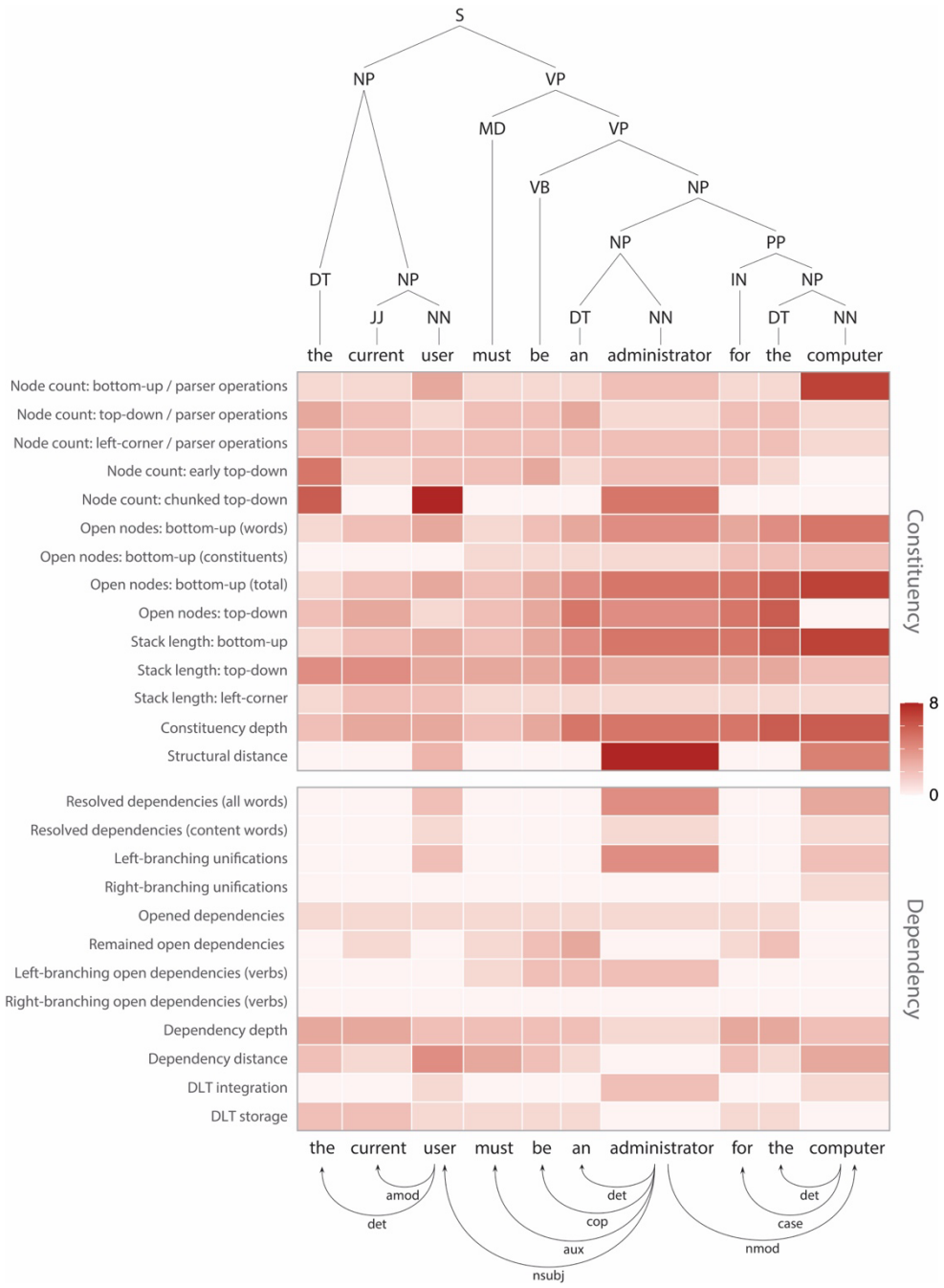

Figure 1. A constituency (top) and dependency (bottom**)** representations of an example sentence, together with the corresponding values of 26 word-level syntactic complexity metrics derived from either representation. Details of the individual metrics are provided in the text (Sections 3.2 and 3.3) and summarized in Tables 1 and 2. The example sentence was adopted from the English LinES treebank (Ahrenberg et al., 2007). The constituency tree was generated with the Charniak-Johnson parser (Charniak & Johnson, 2005), whereas the dependency tree was based on dependency relations included in the treebank. The labels in the constituency tree represent syntactic categories according to the Penn Treebank tagging system: S = sentence, NP = noun phrase, DT = determiner, JJ = adjective, NN = noun, VP = verb phrase, MD = modal, VB = verb, PP = prepositional phrase, and IN = preposition. The dependency labels represent Universal Dependencies grammatical relations: det = determiner, amod = adjectival modifier, nsubj = nominal subject, aux = auxiliary, cop = copula, nmod = nominal modifier, and case = case.

Despite these formal differences, dependency and constituency representations capture similar aspects of syntactic relatedness. On the one hand, constituency relations can be recovered from dependency structures: although dependency representations do not explicitly encode constituents, groups of words connected by directed dependencies often form units that correspond to constituents (Dahl, 1980; Klein & Manning, 2004). For example, in the dependency representation in Figure 1, the head noun *user* has two dependents, *the* and *current*, which together implicitly form the constituent *the current user*, analogous to the corresponding noun phrase in the constituency tree in Figure 1. Conversely, different kinds of dependencies can be derived from constituency structures. For instance, in certain constituency frameworks, the concept c-command plays an important role in explaining relational phenomena (e.g., binding, co-reference; Reinhart, 1983). C-command is an asymmetric, directed relation that specifies the scope of a node in a constituent structure. Although not explicitly represented in constituency, c-command can be recovered from the order in which constituents are hierarchically composed (Collins & Stabler, 2016; Coopmans et al., 2023).

Because constituency and dependency grammars can encode many of the same grammatical relations, they are sometimes described as notational variants (see Nefdt & Baggio, 2024 for discussion). When two grammars are notational variants, their representations are taken to be formally intertranslatable, which means that any structure in one framework can, in principle, be converted into an equivalent structure in the other (Chomsky, 1972; Johnson, 2015). Whether this applies to constituency and dependency grammars depends on the interpretation of equivalence. Constituency and dependency grammars can represent the same sets of sentences, and are therefore equivalent in their weak generative capacity (see Nefdt & Baggio, 2024). However, while dependency structures can be converted into constituency structures, the reverse is not always possible (Gaifman, 1965). Thus, the two frameworks are not fully equivalent in strong generative capacity—constituency and dependency grammars do not have identical empirical consequences (Osborne, 2014).

One example of their non-equivalence concerns the representation of compositional, structure-dependent meaning. This can be illustrated with a simple phrase like *second blue ball,* which admits two interpretations depending on its hierarchical organization (Coopmans et al., 2022; Hamburger & Crain, 1984). Under one analysis, *second* takes scope over the constituent *blue ball*, yielding the meaning “the second of the blue balls”. Under another analysis, *second* and

blue are independent modifiers of *ball*, resulting in the conjunctive meaning "the ball that is blue and second (in position)". Constituency grammars naturally distinguish these readings by generating two constituent structures, namely [second [blue ball]] and [[second] [blue] [ball]]. Under standard word-based dependency representations, however, there would only be a single dependency structure, in which *second* and *blue* are both dependents of the head noun *ball*. As a result, the structural distinction that underlies the semantic ambiguity is not preserved. This example illustrates that the formal non-equivalence of constituency and dependency representations results in distinct empirical consequences.

This discussion about the formal (non-)equivalence between constituency and dependency grammars highlights a productive opportunity for the psycholinguistic and neurolinguistic investigations discussed in this paper. Regardless of whether the two frameworks are ultimately regarded as notational variants, they are formally distinct and therefore make different claims about how grammatical structure is represented, and, by extension, how it is incrementally built during sentence processing. By relating distinct complexity measures derived from constituency- and dependency-based representations to behavioral and neural data, we can directly evaluate the cognitive and empirical adequacy of the respective frameworks. In this way, questions about notational variants can be reframed as questions about how syntactic structure is instantiated in the human brain.

3. Linking syntactic complexity to cognitive load: Computation and memory

In naturalistic investigations of sentence processing, syntactic structures are related to behavioral and neural measures via explicit linking hypotheses, commonly operationalized as word-level syntactic complexity metrics. These metrics are intended to translate abstract properties of linguistic representations into quantitative estimates of the processing cost associated with each word in a sentence. At a general level, such estimates can be understood as reflecting either the extent to which the current syntactic structure must be expanded or modified to integrate an incoming word—thus reflecting the *computational cost* involved in structure building—or the amount of syntactic material held in working memory at that point—thus reflecting *memory cost.* Despite the importance of this approach for connecting syntactic theory to neural data, the cognitive assumptions underlying different metrics are often left implicit, rendering results open to multiple alternative interpretations. In particular, it is often unclear whether observed effects

should be attributed primarily to the cost of performing syntactic computations or the cost of holding partial syntactic structures in memory.

To address this issue, we introduce a taxonomy of word-level syntactic complexity metrics that is grounded in the cognitive distinction between computation- versus memory-related aspects of syntactic processing.[1] This distinction builds on a long-standing tradition in psycholinguistics that links processing cost to the properties of syntactic structure (Abney & Johnson, 1991; Fodor et al., 1974; Ford et al., 1982; Frazier, 1985; Gibson, 1998; Johnson-Laird, 1983; Kaplan, 1972; Kimball, 1973; Miller & Chomsky, 1963; Pritchett, 1992; Yngve, 1960). In this literature it was recognized that partial, incrementally constructed linguistic structures must be held in memory in order to support the integration of subsequent input.

In the following sections, we review the most widely used word-level syntactic complexity metrics. We evaluate whether a given metric primarily targets computation or memory, and we further distinguish metrics according to the grammatical framework on which they are based (constituency versus dependency). Because studies rarely make explicit the cognitive interpretation of their metrics, the classification into computation and memory reflects our own interpretation of the theoretical analysis of each metric.

As we are primarily interested in explaining the (neural) dynamics of incremental syntactic structure building, our review is focused on word-level complexity metrics. However, several studies have employed relevant sentence-level measures of syntactic complexity, such as mean dependency distance (Liu, 2008), maximal local nonterminal count (Frazier, 1985), average syntactic rule frequency (Rezaii et al., 2022), and various scales derived from quantifying syntactic structures (Agmon et al., 2024). Only when a sentence-level measure was derived from an underlying word-level metric, the word-level metric is included in the review.

3.2 Measures of computation

As each word is encountered, the parser incrementally updates the evolving syntactic representation by integrating the word into the existing parse. Computation metrics quantify the processing cost associated with these structure-building operations. Different metrics

[1] A similar distinction was proposed by Gibson (2000), who labeled these components *integration* and *storage*, respectively. However, as we will discuss in Section 3.3, Gibson's measure of *integration* quantifies retrieval-related demands on working memory and is therefore fundamentally distinct from our concept of computation. Moreover, *storage* measure reflects an anticipatory form of storage, whereas our concept of *memory* is broader (see Section 3.3).

operationalize structure-building cost in different ways, depending on the grammatical formalism and parsing strategy they assume.

3.2.1. Constituency metrics

In early psycholinguistic models, constituency-based measures of computation quantified the amount of constituent structure built at each word (Frazier, 1985; Miller & Chomsky, 1963). Frazier (1985) proposed an algorithm in which each word is assigned a score corresponding to the number of phrasal nodes the word introduces into the evolving structure. This *Frazier score* laid the foundation for a family of ***node count*** metrics, which have been widely adopted in recent neuroimaging work (Bhattasali et al., 2019; Brennan & Pylkkänen, 2017; Brennan et al., 2012, 2016; Cai et al., 2026; Coopmans et al., 2022, 2025; Giglio et al., 2024; Gilliams et al., 2025; Li & Hale, 2019; Lopopolo et al., 2021; Nelson et al., 2017; Reddy & Wehbe, 2021; Slaats et al., 2024; Stanojević et al., 2023; Weissbart & Martin, 2024).[2] Node count estimates computation as the number of nodes that must be constructed to incorporate each incoming word into the evolving constituency structure.

Nodes can be constructed following different parsing strategies (see Hale, 2014). A ***bottom-up*** strategy postulates a constituent node only after all of its daughter nodes are available, a ***left-corner*** strategy builds a node once its left-most daughter node is available, and a ***top-down*** strategy immediately postulates all nodes (up to the root node) needed to attach the upcoming word to the full structure (see Table 1 for an overview). Although all three strategies ultimately construct the same constituency tree, the time points at which nodes are added to the structure differ, meaning that they make different predictions about the dynamics of structure building. More specifically, the parsing methods differ in the completeness of the input they require to build structure, with the top-down method building structure maximally eagerly and predictively (using uncertain and incomplete input), while the bottom-up method requires complete input and is therefore non-predictive. The left-corner method is mildly predictive, as it allows limited prediction while still requiring partial bottom-up evidence.

Node count is typically derived from context-free phrase-structure grammars. However, computation metrics can be derived from alternative constituency formalisms as well, such

[2] Some studies refer to node count as *number of closed phrase structures* (Lopopolo et al., 2021), *number of nodes closing* (Gwilliams et al., 2025; Nelson et al., 2017), *close* (Weissbart & Martin, 2024) or *closing level* (Cai et al., 2026). These metrics are identical to bottom-up node count and therefore not included as unique metrics.

Combinatorial Categorial Grammars (CCGs; Kajikawa et al., 2024; Stanojević et al., 2023). In contrast to Context-Free Grammars (CFGs), CCG is a mildly context-sensitive formalism. It can capture constructions such as crossing serial dependencies, which fall outside the expressive capacity of CFGs. Importantly, CCG supports flexible constituency, which allows multiple semantically equivalent derivations of the same sentence. Parsing can therefore proceed more eagerly, with syntactic and semantic composition taking place before the corresponding constituents would become available in a context-free phrase-structure grammar. This makes CCG particularly well suited for modeling word-by-word sentence processing (Steedman, 2000; Stanojević et al., 2023). Accordingly, ***CCG step count*** operationalizes computation as the number of parsing operations performed at each word. Parsing can proceed using a left-branching strategy, in which operations (*Shift*, *Reduce*) are applied as soon as semantic composition is grammatically licensed, or using a revealing strategy, which introduces an additional transformation step (*Reveal*) that converts the partial parse into a semantically equivalent structure better suited to handling right adjuncts (Stanojević et al., 2023). Expanding this approach, individual composition steps can be classified according to the combinatorial rule involved—such as *Function application*, *Function composition*, or *Type raising*—and counted separately (Kajikawa et al., 2024). Note that while *node count* and *CCG step count* are derived from different grammatical formalisms, both operationalize syntactic computation by counting the structure-building operations associated with each incoming word.

3.2.2. Dependency metrics

In dependency grammar, syntactic structure is formed by establishing head–dependent relations between words. Accordingly, several computation metrics derived from dependency representations estimate the cost of structure building in terms of the number of dependency relations resolved at each word. For instance, the *number of left-hand dependency relations* – referred to as ***resolved dependencies (all words)*** in Table 1 – counts the number of dependencies a word has with preceding words (Lopopolo et al., 2021). It estimates the operations required to integrate that word into the unfolding dependency structure. A related metric, called ***resolved dependencies (content words)***, counts the number of left-hand dependency relations involving content words (Zioga et al., 2023). Dependencies involving function words were excluded because their contribution to high-level semantic comprehension is limited.

Notably, neither metric distinguishes the grammatical direction of the dependency—that is, whether the dependent precedes or follows the head, and thus whether the word resolving the dependency is a head or a dependent. Kapteijn and Hintz (2021) included this information in two of their metrics. In their terminology, the number of ***left-branching unifications*** reflects the number of leftward dependencies resolved by the head of the dependency, whereas the number of ***right-branching unifications*** reflects the number of rightward dependencies that are resolved by the dependent. The motivation for including both metrics is that the cost of resolving a dependency may differ depending on whether the head precedes the dependent (as in right-branching unifications) or follows it (as in left-branching unifications).

Table 1. Overview of all metrics of computation based on constituency and dependency grammars.

Constituency grammar				
Metric	**Syntactic representation**	**Details**	**Language**	**Reference**
Node count	Context-free grammar	Bottom-up: Number of nodes constructed at a word according to a bottom-up parsing strategy.	English	Brennan et al. (2012, 2016); Nelson et al. (2017)[3], Bhattasali et al. (2019); Li & Hale (2019); Lopopolo et al. (2021), Reddy & Wehbe (2021); Stanojević et al. (2023); Giglio et al. (2024); Gwilliams et al. (2024); Cai et al. (2026)
			Dutch	Weissbart & Martin (2024)
		Top-down: Number of nodes constructed at a word according to a top-down parsing strategy.	English	Frazier (1985)[4]; Brennan et al. (2016); Giglio et al. (2024); Stanojević et al. (2023)
		Left-corner: Number of nodes constructed at a word according to a left-corner parsing strategy.	English	Brennan & Pylkkänen (2017)
		Early top-down: A temporally shifted version of top-down node count, in which the count associated with word $n + 1$ is assigned to word n.	English	Giglio et al. (2024)
		Chunked top-down: Total number of constituency nodes constructed in a top-down manner within each dependency-defined chunk, counted from a dependency head up to and including the next dependency head. The resulting number is assigned to the leftmost word of the dependency.	English	Giglio et al. (2024)

[3] In Nelson et al. (2017), syntactic trees were generated with custom MATLAB code. The authors describe their parser as "[relying] on a simplified version of X-bar theory", but structural details on this implementation are sparse. Since their examples contain no empty categories, we classified it as a context-free grammar in this overview.

[4] In Frazier (1985), phrasal nodes are introduced according to a top-down tree traversal, with the root node weighted as 1.5 rather than 1 and terminal nodes excluded from the count. Because these differences result in values that very closely approximate *top-down node count*, we did not include the *Frazier score* as a separate metric in the clustering analysis (Section 4).

	X-bar theory	Bottom-up: Number of nodes constructed at a word according to a bottom-up parsing strategy. Nodes counted at the position of empty categories were added to the node count of the subsequent word.	Dutch	Slaats et al. (2024); Coopmans et al. (2025)
		Top-down: Number of nodes constructed at a word according to a top-down parsing strategy. Nodes counted at the position of empty categories were added to the node count of the subsequent word.	Dutch	Slaats et al. (2024); Coopmans et al. (2025)
		Left-corner: Number of nodes constructed at a word according to a left-corner parsing strategy. Nodes counted at the position of empty categories were added to the node count of the subsequent word.	Dutch	Coopmans et al. (2025)
	Minimalist grammar	Bottom-up: Number of nodes constructed at a word according to a bottom-up parsing strategy, with empty nodes included in the count.	English	Brennan et al. (2016); Li & Hale (2019)
		Top-down: Number of nodes constructed at a word according to a top-down parsing strategy, with empty nodes included in the count.	English	Brennan et al. (2016)
Step count	Context-free grammar[5]	Bottom-up: Number of *Shift* and *Reduce* operations applied at a word.	English, French	Nelson et al. (2017)
		Top-down: Number of *Scan* and *Expand* operations applied at a word.	English, French	Nelson et al. (2017)
		Left-corner: Number of *Shift* and *Project* operations applied at a word.	English, French	Nelson et al. (2017)

[5] The parsing operations adopted by Nelson et al. (2017) are a linear transformation of node count metrics, because the *Shift* and *Scan* operations contribute one operation per word, whereas *Reduce*, *Expand*, and *Project* operations are determined by the number of nodes constructed according to a bottom-up, top-down, and left-corner strategy, respectively. Thus, if *bottom-up node count* for a given word is 4, the corresponding number of bottom-up parsing operations is 1 (*Shift*) + 4 (*Reduce*) = 5. Because these parsing operations are perfectly correlated with node count across words, they are treated together in Figure 1 and the clustering analysis (Section 4).

	Combinatorial Categorial Grammar (CCG)	CCG left-branching: Number of *Shift* and *Reduce* operations applied at a word.	English	Stanojević et al. (2023)
		CCG revealing: Number of *Shift*, *Reduce* and *Reveal* operations applied at a word. *Reveal* transforms a partial tree into a semantically equivalent structure.	English	Stanojević et al. (2023)
		Number of *Function application* operations applied at a word.	English, Japanese	Kajikawa et al. (2024)
		Count of *Type raising* operations applied at a word.	English, Japanese	Kajikawa et al. (2024)
		Count of *Function composition* operations applied at a word.	English, Japanese	Kajikawa et al. (2024)
		Number of *Function application*, *Type raising*, and *Function composition* operations applied at a word.	English, Japanese	Kajikawa et al. (2024)
Dependency grammar				
Metric	**Syntactic representation**	**Details**	**Language**	**Reference**
Resolved dependencies (all words)	Alpino	Number of dependencies that are resolved at a word.	Dutch	Lopopolo et al. (2021)
Resolved dependencies (content words)	Universal Dependencies	Number of dependencies that are resolved at a word, excluding dependencies involving function words.	Dutch	Zioga et al. (2023)
Left-branching unifications	Alpino	Number of left-branching dependencies that are resolved by a head.	Dutch	Kapteijns & Hintz (2021)

Right-branching unifications	Alpino	Number of right-branching dependencies that are resolved by a dependent.	Dutch	Kapteijns & Hintz (2021)

Note 1. The column *Syntactic representation* refers to the syntactic representation used in the original reference. For constituency-based metrics, this typically reflects the grammatical formalism (e.g., context-free grammar, Minimalist Grammar) and the annotation scheme used to encode phrase structure (e.g., Penn Treebank conventions). For dependency-based metrics, it refers to the dependency representation or framework (e.g., Stanford Dependencies, Universal Dependencies, Alpino/Lassy) or, when the original study reports only the software, the dependency parser used (e.g., Frog, MiniPar). Different dependency representations may differ in head selection, dependency relations, and other structural analyses.

Note 2. In the context of this review, the relevant distinction between phrase structures derived from context-free grammars, X-bar theory, and minimalist grammars concerns the level of syntactic detail they encode. Trees derived from context-free grammars may contain either binary or n-ary branching structures. X-bar structures are uniformly binary branching and include empty categories associated with movement dependencies. Structures derived from minimalist grammars are also binary branching and contain empty nodes indicating movement dependencies, but additionally contain unary (non-branching) projections. These representational differences affect the number and distribution of nodes in the resulting trees and therefore strongly influence node count metrics.

3.3 Measures of memory

Memory metrics quantify memory demands arising from incremental structure-building operations. Depending on the metric, these demands may arise from syntactic elements that have been predicted but not yet encountered (an anticipatory form of storage), from syntactic elements that have been encountered but not yet fully integrated (maintenance of unfulfilled syntactic commitments), or from retrieving words from working memory in order to resolve a syntactic dependency. In the following sections, we refer to these three forms of memory as *anticipation, maintenance*, and *retrieval*, respectively.

3.3.1. Constituency metrics

Constituency-based memory metrics typically quantify the amount of constituent structure that must remain accessible in working memory during incremental structure building (see Table 2 for an overview). One of the earliest explicit measures of memory cost was introduced by Yngve (1960), who developed a model of sentence production in which the amount of syntactic material that must be held in memory predicts processing cost. The resulting metric—known as *Yngve score* or *Yngve depth*—is computed by assigning the value 0 to the right branch of each constituent and the value 1 to the left branch, and then summing the values of the branches along the path from the root of the tree to the current word. Under this metric, nodes that are hierarchically deeper on the left branch receive a higher score, so memory cost is determined by the number of left branches that dominate a given word. The rationale is that whenever the derivation proceeds along a left branch, the speaker anticipates the eventual realization of the corresponding right branch. Temporary storage of these unfulfilled syntactic commitments imposes a burden on working memory.

Building on this seminal work, related memory metrics have been proposed based on the number of ***open nodes*** maintained in memory during incremental structure building. The *open-nodes* metric reflects the total number of constituent nodes that have been opened by the parser but not yet closed at the current word. Unlike *Yngve score*, however, whether this metric reflects maintenance or anticipation depends on the parsing strategy that is assumed. When a bottom-up strategy is used (Nelson et al., 2017), *open nodes* captures the temporary storage of directly encountered but not yet integrated syntactic structure. This corresponds to maintaining partially constructed constituents (their left corner) until the remaining material (their right corner) has been

encountered. In contrast, when nodes are opened according to a top-down parsing strategy (Giglio et al., 2024; Gwilliams et al., 2025), *open nodes* captures anticipated constituents, including incomplete nodes dominating the current word. Consequently, the top-down version predicts a larger memory load than the bottom-up version because it includes anticipated as well as maintained constituents.

A more explicit, parser-based operationalization of memory load is obtained by tracking the number of nodes stored in a parser's stack, yielding the metric ***stack length*** (Nelson et al., 2017). Under a bottom-up parsing strategy, *stack length* closely mirrors *open nodes*, because both quantify the amount of partially constructed constituent structure that must be maintained until it can be completed.[6] Alternative parsing strategies (top-down and left-corner) produce different, more anticipatory storage profiles and therefore yield different values of stack length.[7]

The metric ***constituency depth*** is computed as the number of nodes along the path from the current leaf node to the root of the tree structure (Gwilliams et al., 2025; Weissbart & Martin, 2024). Like *Yngve score*, constituency depth increases with the depth of syntactic embedding. However, whereas *Yngve score* counts only left branches corresponding to unresolved syntactic commitments, *constituency depth* counts all constituent nodes along the path to the root, including nodes dominated by right branches. The underlying assumption is that integrating the incoming word requires maintaining the currently active hierarchical context in working memory. However, *constituency depth* operationalizes memory differently than the preceding metrics. Unlike *stack length* and *open nodes*, *constituency depth* is not derived from a parser state; it does not explicitly encode unfinished or predicted constituents. Instead, it treats hierarchical embedding as a proxy for the amount of syntactic context that must remain accessible during incremental structure building. Consequently, *constituency depth* does not distinguish between maintenance and anticipatory forms of storage. The relative contribution of each depends on the extent to which the

[6] Nelson et al. (2017) measured *stack length* at each word before that word was integrated into the syntactic structure. As a result, the sentence-final word retains a relatively high count of *open nodes*, whereas in a simple bottom-up operationalization the count of *open nodes* is zero at the final word. This difference reflects a particular implementation choice with empirically testable consequences; namely, a predicted neural increase in response to sentence-final words.

[7] Under the top-down parsing strategy, the length of the stack for a binary branching tree is equivalent to *Yngve score*, as both quantify the number of expected right branches along the path from the current word to the root. Because we derived all constituency metrics from binary trees, we grouped *Yngve score* with *stack length: top-down* and did not include it as a separate metric in the clustering analysis (Section 4).

constituents dominating the current word correspond to partially realized or projected structure. In predominantly left-branching regions of a tree, *constituency depth* primarily reflects anticipatory storage, similarly to the *Yngve score*. For example, in [[[John's friend's] dog's] food], the depth of *John* is 3, corresponding to nodes that are anticipated after processing that word. In predominantly right-branching regions, however, *constituency depth* primarily reflects maintenance of partially constructed structure before processing that word. Thus, in [John [liked [Bill's dog]]], the depth of *dog* is also 3, but this value corresponds to nodes that had to be maintained in memory in order to integrate that word into the existing structure.

Finally, ***structural distance*** combines information from constituency and dependency representations (Baumann, 2014; Li & Hale, 2019). It is defined as the total number of non-terminal constituency nodes separating two words that stand in a dependency relation. This number is assigned to the rightmost word of the dependency. When a word resolves multiple dependencies, the corresponding structural distances are summed. For example, in Figure 1, *user* is the rightmost word in two dependencies, one with *current* and one with *the*. The total structural distance is therefore 3, corresponding to the sum of the non-terminal nodes intervening between *user* and its two dependents. This metric captures a retrieval-like process, and is based on the assumption that memory load increases with the amount of hierarchical structure that separates two words when their dependency is resolved. Dependencies spanning more constituent structure are assumed to impose greater demands on memory.

3.3.2. Dependency metrics

Dependency-based memory metrics quantify the amount of dependency information that must remain accessible in working memory during incremental parsing (see Table 2 for an overview). Zioga et al. (2023) introduced two metrics that count unresolved dependency relations, which capture the idea that open dependencies must be maintained in working memory until they are resolved. ***Opened dependencies*** counts the number of dependencies initiated at the current word, whereas ***remained open dependencies*** counts the number of previously initiated dependencies that remain unresolved. Neither metric distinguishes the directionality of the dependency (i.e., whether the dependent precedes or follows its head), though this distinction is captured by two related measures. ***Left-branching open dependencies*** quantifies unresolved left-branching dependencies, in which the dependent is on the left-hand side of the head, whereas ***right-branching open***

dependencies quantifies unresolved right-branching dependencies, in which the dependent is on the right-hand side of the head (Uddén et al., 2022).[8] These measures differ in the information they store, thereby potentially capturing different degrees of anticipatory processing: *left-branching open dependencies* involves maintaining dependents until their head is encountered, whereas *right-branching open dependencies* involves maintaining heads while anticipating their dependents. Kapteijns & Hintz (2021) proposed two related metrics, *left-branching complexity* and *right-branching complexity*, which are computed by summing the metrics of *unifications*, *opened dependencies* and *remained open dependencies* for left- and right-branching relations, respectively. Because the present review focuses on metrics that can isolate individual cognitive processes, these composite measures are not included in the quantitative analysis presented in Section 4.

Two additional memory metrics have close counterparts among constituency-based metrics. First, ***dependency depth*** is computed as the number of dependency relations along the path from the current word to the root of the dependency tree (Cai et al., 2026). Like *constituency depth*, *dependency depth* treats hierarchical embedding as a proxy for the amount of syntactic context that must remain accessible during incremental structure building. And like *constituency depth*, it captures different forms of memory-related components depending on the local tree configuration. When most heads along the path from a word to the root are to the right of that word, the *dependency depth* of that word primarily reflects *anticipatory* storage. Conversely, when most heads lie to the left of that word, *dependency depth* primarily reflects the *maintenance* of unresolved dependencies. Second, ***dependency distance*** reflects the linear distance between a dependent and its head (Futrell et al., 2015; Liu, 2008; Liu et al., 2017). Like *structural distance*, it is based on the assumption that memory load increases with the amount of syntactic material separating two words that stand in a dependency relation. However, the two metrics differ in where the resulting cost is assigned. *Structural distance* assigns cost to the rightmost word of the dependency, whereas *dependency distance* assigns cost to the dependent. And because *dependency distance* does not distinguish left- from right-branching dependencies, the dependent can be

[8] Note that Uddén et al. (2022) label these metrics *left-branching complexity* and *right-branching complexity*, respectively. We use the labels *left-branching open dependencies* and *right-branching open dependencies* to distinguish them from the composite metrics of Kapteijns & Hintz (2021), which use the same labels but combine multiple dependency measures.

located to the right or to the left of the head. This metric therefore conflates anticipation (for left-branching dependencies) and retrieval (for right-branching dependencies).

Dependency Locality Theory (DLT) is a memory-based theory of sentence processing in which the cost of integrating two elements increases with the distance between them, where distance is defined in terms of the complexity of the intervening discourse structure (Gibson, 1998, 2000). Accordingly, the ***DLT integration cost*** of a word is computed as the number of discourse referents (i.e., nouns and finite verbs) that were introduced between that word and the leftmost word with which it forms a dependency (Demberg & Keller, 2008; Gibson, 1998, 2000; Shain et al., 2016, 2022). The assumption is that resolving a dependency requires retrieving the earlier element from working memory. Because intervening discourse referents interfere with the stored representation, retrieval becomes more costly when more discourse referents intervene. Although DLT was originally formulated within a constituency-based framework, locality is defined linearly (in terms of intervening discourse referents), allowing *DLT integration* to be implemented over dependency representations as well (Gibson, 2025).

For word-level analyses of naturalistic language comprehension, a limitation of *DLT integration* is that it assumes retrieval costs to arise for nouns and verbs only. The majority of all words receive a predicted integration cost of zero, limiting the metric's predictive range. Furthermore, the precise integration weights are not specified by the theory. Although these weights are often used as heuristic approximations that instantiate a general pattern, it remains unclear to what extent specific assumptions matter when predicting behavioral and neural responses. Systematic comparisons between alternative implementations (e.g., distinguishing how finite and nonfinite verbs are weighted) are therefore needed (see Demberg & Keller, 2008; Shain et al., 2016, 2022).

The second DLT metric is called ***DLT storage cost***. It reflects the minimal number of syntactically mandatory elements that are required to complete the current input as a grammatical sentence (Gibson, 1998, 2000; Shain et al., 2022). For example, in the sentence "children eat cookies", the storage cost at "children" is 1, because the subject requires at least one syntactic element—an intransitive verb—to form a grammatical sentence. The storage cost at the verb "eat" is also 1 because a transitive verb needs a complement, whereas the storage cost at "cookies" is 0 because no additional syntactic elements are required to complete the sentence. *DLT storage*

differs from *DLT integration* in being a forward-looking measure: whereas *DLT integration* reflects the difficulty of retrieving previously introduced elements, *DLT storage* reflects the amount of syntactic structure that is predicted to be required to complete the current input. It therefore captures anticipatory storage of predicted dependencies.

Table 2. Overview of all metrics of memory based on constituency and dependency grammars.

Constituency grammar				
Metric	**Syntactic representation**	**Details**	**Language**	**Reference**
Open nodes	Context-free grammar	Words: Number of lexical nodes that have been opened (up to and including the current word) but not yet closed at a word.	English, French	Nelson et al. (2017)
		Constituents: Number of phrasal nodes that have been opened (up to and including the current word) but not yet closed at a word.	English, French	Nelson et al. (2017)
		Total: Number of nodes that have been opened (up to and including the current word) but not yet closed at a word.	English	Nelson et al. (2017)
		Top-down: Number of nodes constructed under a top-down parsing strategy but not under a bottom-up strategy. Identical to top-down node count minus bottom-up node count at a word.	English	Giglio et al. (2024); Gwilliams et al. (2025)
Stack length	Context-free grammar	Bottom-up: Number of nodes stored in the parser's stack according to bottom-up parsing strategy.	English, French	Nelson et al. (2017)
		Top-down: Number of nodes stored in the parser's stack according to a top-down parsing strategy.	English, French	Yngve (1960); Nelson et al. (2017)
		Left-corner: Number of nodes stored in the parser's stack according to a left-corner parsing strategy.	English, French	Nelson et al. (2017)
Constituency depth	Context-free grammar	Number of nodes along the path from the current leaf node to the root of the structure.	Dutch	Weissbart & Martin (2024)
			English	Gwilliams et al. (2025); Cai et al. (2026)

Structural distance	Context-free grammar	Total number of phrasal nodes separating two words that stand in a dependency relation. The resulting distance is assigned to the rightmost word of the dependency.	English	Baumann (2014); Li & Hale (2019)
Dependency grammar				
Metric	**Syntactic representation**	**Details**	**Language**	**Reference**
Opened dependencies	Universal Dependencies	Number of dependencies that are opened at a word.	Dutch	Zioga et al. (2023)
Remained open dependencies	Universal Dependencies	Number of dependencies that are already open but remain unresolved at a word.	Dutch	Zioga et al. (2023)
Left-branching open dependencies (verbs)	Alpino	Number of simultaneously open left-branching dependencies at a word. Restricted to dependencies involving a verb.[9]	Dutch	Uddén et al. (2022)
Right-branching open dependencies (verbs)	Alpino	Number of simultaneously open right-branching dependencies at a word. Restricted to dependencies involving a verb.	Dutch	Uddén et al. (2022)
Dependency depth	AllenNLP Dependency parser	Number of dependencies along the path from the current word to the root of the sentence.	English	Cai et al. (2026)
Dependency distance	Language-specific treebanks	Absolute difference in linear position between a word and its head. The number is assigned to the dependent.	Cross-linguistic	Liu (2008); Liu et al. (2017)
DLT integration	MiniPar	Number of discourse referents that intervene in a backward-looking dependency.	English	Demberg & Keller (2008); Shain et al. (2016, 2022)

[9] The two *open dependencies* metrics (Uddén et al., 2022) are conceptually similar to *remained open dependencies* (Zioga et al., 2023) because they capture the memory demands associated with keeping track of dependencies that have not yet been resolved. However, they assume these demands to occur at different time points. First, *open dependencies* is increased at the word opening the dependency, whereas *remained open dependencies* is increased only for previously opened but non-resolved dependencies. Second, *remained open dependencies* is decreased at the word resolving the dependency, whereas *open dependencies* is decreased at the next word. These somewhat arbitrary assumptions about the order in which dependents are added to and removed from a memory stack lead to different predictions about when storage demands are high.

DLT storage	Not specified	Minimal number of syntactically mandatory elements required to form a grammatical sentence.	English	Shain et al. (2022)

Note. The column *Syntactic representation* refers to the syntactic representation used in the original reference. For constituency-based metrics, this typically reflects the grammatical formalism (e.g., context-free grammar, Minimalist Grammar) and the annotation scheme used to encode phrase structure (e.g., Penn Treebank conventions). For dependency-based metrics, it refers to the dependency representation or framework (e.g., Stanford Dependencies, Universal Dependencies, Alpino/Lassy) or, when the original study reports only the software, the dependency parser used (e.g., Frog, MiniPar). Different dependency representations may differ in head selection, dependency relations, and other structural analyses.

4. Hierarchical clustering of metrics

As evident from Section 3, previous behavioral and neuroscientific studies have employed a wide range of syntactic complexity metrics, while often making insufficiently explicit which cognitive processes they reflect. We proposed a taxonomy that organizes these metrics according to their underlying grammatical representations and the cognitive processes they are likely to capture. In this section, we quantitatively evaluate the proposed taxonomy of computation and memory by examining whether metrics assigned to the same cognitive component covary, both within and across grammatical formalisms. If they do, that would provide support for using the computation–memory distinction as an organizing framework for interpreting results in the neuroimaging literature (see Section 5).

We computed 26 metrics on a sample of 100 sentences drawn from the development set of the English LinES Parallel Treebank (Ahrenberg, 2007) of the Universal Dependencies project (de Marneffe et al., 2021; Nivre et al., 2016).[10] Sentences were selected sequentially from the beginning of the development set, skipping titles, sentence fragments, URLs and sentences containing fewer than four words. We also excluded a small number of entries containing multiple orthographic sentences represented as a single dependency tree (via a *parataxis* relation), as these could not be directly compared with the constituency parses that would analyze them as separate sentences. The final sample of 100 sentences consisted of 2,044 words, with sentence lengths ranging from 4 to 55 words (M = 20.4, SD = 8.3).

Constituency and dependency metrics were computed using custom code. For the constituency metrics, sentences were first parsed with the Charniak–Johnson parser (Charniak & Johnson, 2005) to generate binary phrase-structure trees consistent with a context-free grammar. The metrics were derived from these parse trees. For the dependency metrics, we relied on the gold-standard dependency annotations provided with the corpus and computed all metrics automatically, with the exception of *DLT storage*. This metric was non-trivial to compute automatically and hence computed manually. We relied on the definition provided by Shain et al. (2022, pp. 7417-7418), where *DLT storage* corresponds to the minimum number of additional syntactic heads that is required to complete the sentence in a grammatical way. For all metrics, we

[10] https://github.com/UniversalDependencies/UD_English-LinES/blob/writer/en_lines-ud-dev.conllu

closely followed the definitions and procedures described in the original studies (see Tables 1 and 2) and verified the output of our code against examples provided in these studies. To ensure reproducibility and facilitate the re-use of these metrics, all code is publicly available via two Google Colab notebooks.[11]

To examine the quantitative relationships among the different metrics, we applied a hierarchical clustering analysis. We computed pairwise Pearson correlation coefficients (r) across all 2,044 standardized word-level scores. Hierarchical agglomerative clustering was then performed using correlation distance ($1-r$) and complete linkage, in which the distance between two clusters is defined as the largest pairwise distance between any member of one cluster and any member of another. The results are visualized using dendrograms shown in Figures 2 and 3. To evaluate different dimensions of the proposed taxonomy, clustering analyses were performed separately for constituency- and dependency-based metrics (Figures 2A-B), for memory- and computation-related metrics (Figures 2C-D), and for the complete set of metrics (Figure 3).

When constituency-based and dependency-based metrics are analyzed separately, computation- and memory-related metrics appear to form distinct clusters (Figures 2A-B). This finding provides quantitative support for using the computation–memory distinction as an organizing framework when reviewing the neural correlates of syntactic complexity (see Section 5). In contrast, when all metrics are analyzed together, this distinction no longer emerges as an organizing component (Figure 3). This suggests that the underlying grammatical formalism shapes the overall organization of the quantitative relationships among syntactic complexity metrics. Metrics intended to capture the same cognitive process may be related more strongly when derived from the same grammatical formalism than when derived from different formalisms.

[11] Constituency notebook: https://colab.research.google.com/drive/1DDLi0CALAmQq7F6aeRmVqDtz624a93-j?usp=sharing
Dependency notebook: https://colab.research.google.com/drive/14GgSMB9veOKNpReEWBow1geZM_ZhZr_V#scrollTo=zPIO-JraJZ8t

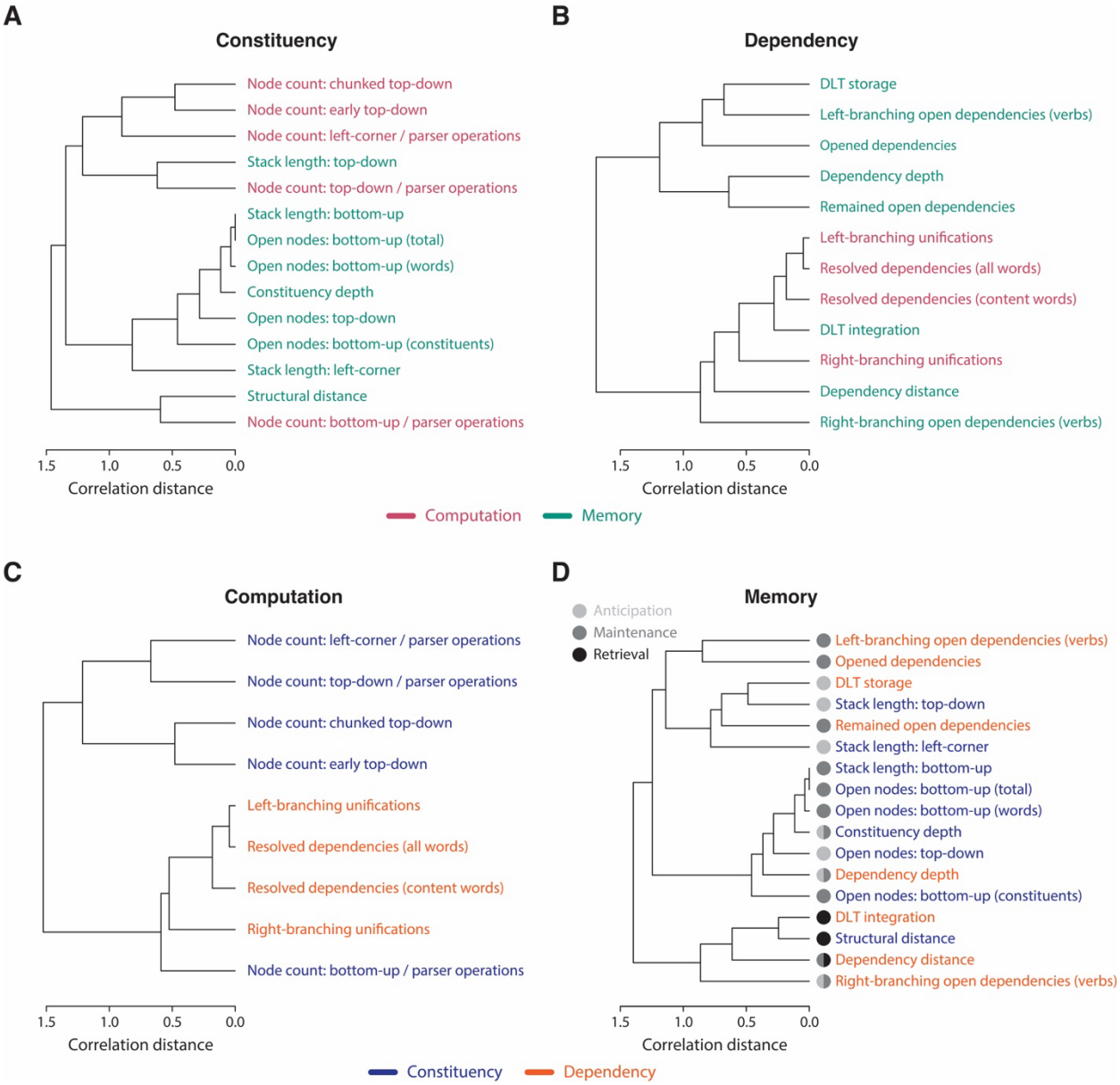

Figure 2. Hierarchical clustering of all metrics, grouped by constituency **(A)** versus dependency **(B)** and by computation **(C)** versus memory **(D)**. The memory metrics in **(D)** are additionally color-coded by the subtype of memory they reflect. The lower the correlation distance between two metrics, the higher their correlation and the stronger their clustering.

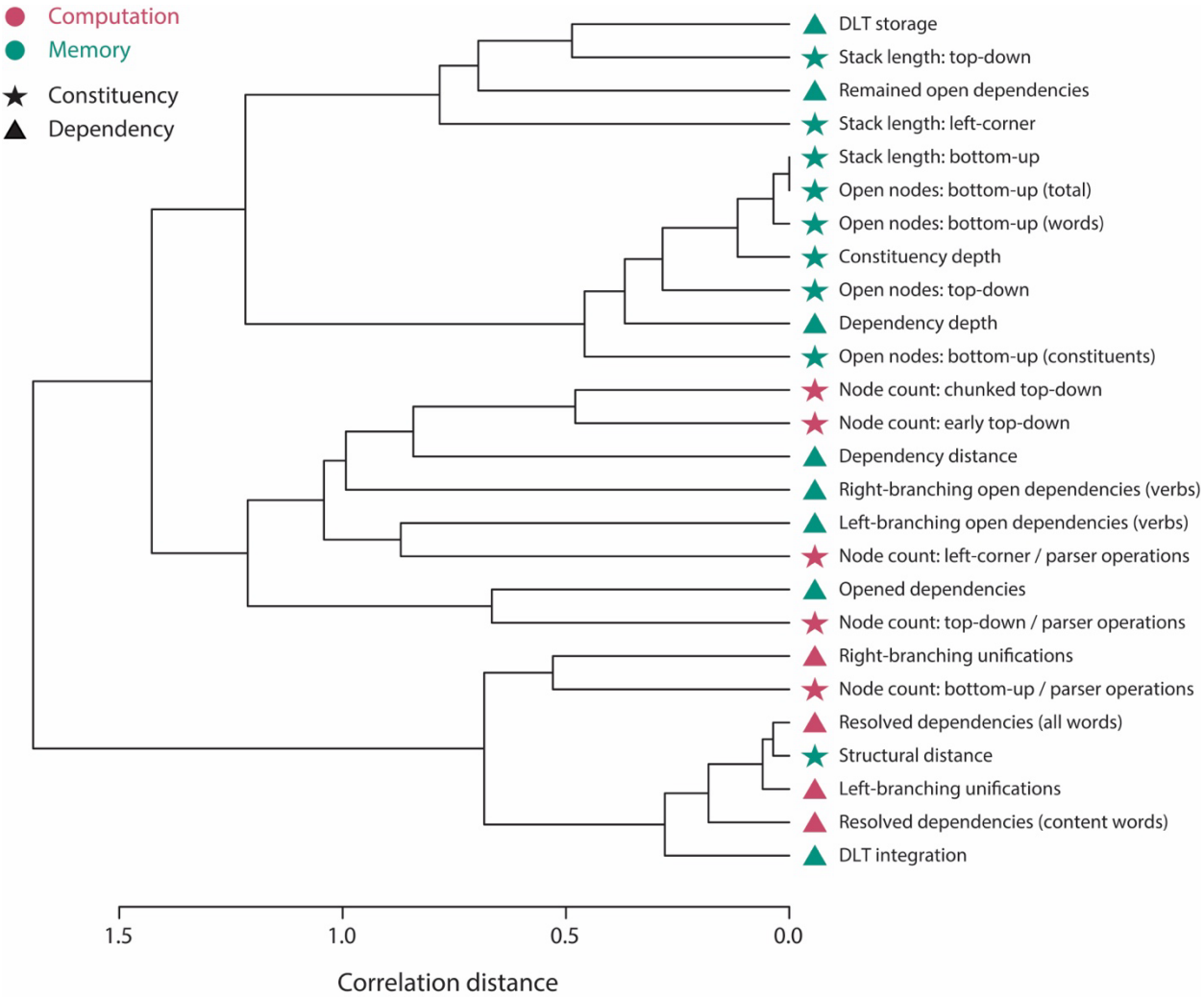

Figure 3. Hierarchical clustering of all metrics combined. The contrast between computation- and memory-related metrics is color coded. The contrast between constituency- and dependency-related metrics is indicated by means of a shape. The lower the correlation distance between two metrics, the higher their correlation and the stronger their clustering.

The computation–memory clustering pattern is clearly illustrated in the constituency dendrogram (Figure 2A). The *node count* metrics, which quantify computation cost, group together (dark pink) in a cluster separate from most memory metrics (cyan). There are two exceptions to this pattern. First, *bottom-up node count* clusters with *structural distance*—a memory metric—rather than with the other computation metrics. This pattern is consistent with the definition of *structural distance*, which estimates retrieval effort by computing, via bottom-up traversal, all nodes along the path between two dependency heads (Li & Hale, 2019). *Structural distance* thus combines a *bottom-up node-count* computation with a retrieval-based measure of memory demand. Second, *top-down stack length* clusters with the computation metrics rather than with the other memory metrics. In a

top-down traversal, the stack consists of syntactic nodes that have been predicted before the supporting input is encountered. *Top-down node count* measures how many constituents are predicted (via *Expand*) before a word is encountered. *Top-down stack length* measures how many of those predicted constituents remain active (i.e., are still waiting to be completed) immediately before the word is encountered. Thus, the stack is essentially the residual of previous *Expand* operations, possibly explaining why these two metrics are correlated: A parser that predicts many constituents performs many *Expand* operations and also maintains many predicted constituents on the stack. The distinction between computation and memory is therefore less pronounced for top-down parsing than for the other parsing strategies.

Notably, *bottom-up node count* does not cluster with *left-corner node count*, despite the fact that both contain an integrative component. A possible explanation is that, unlike *bottom-up node count*, the other node-count metrics all contain an anticipatory component as well: *top-down* and *left-corner* parsing strategies posit syntactic nodes before their complete input has been encountered. This interpretation is further supported by the clustering analysis within all computation metrics (Figure 2C), in which *bottom-up node count* clusters with the integrative dependency metrics, suggesting that it is more closely related to input-driven integration than to anticipation-based computation.

A similar separation between computation- and memory-related metrics is observed for the dependency metrics, although the division is not as sharp (Figure 2B). While computation metrics generally cluster together, as do memory metrics, they are also mixed together. Notably, three memory metrics (*DLT integration*, *dependency distance*, and *right-branching open dependencies*) cluster with the computation metrics rather than the other memory metrics. This pattern can be partially explained by the proposed sub-taxonomy within memory. Specifically, two of these three memory metrics—*DLT integration* and *dependency distance* (for rightward dependencies)—index retrieval-related aspects of syntactic processing, suggesting that retrieval forms a distinct subtype of cognitive demand. The placement of the third metric in this cluster, *right-branching open dependencies*, is less straightforward. This metric reflects both anticipation and maintenance, as it counts dependencies whose heads have been encountered but whose dependents have not yet appeared, thereby indexing both the prediction of upcoming dependents and the maintenance of

unresolved dependencies. Consistent with this interpretation, it also exhibits the weakest correlation with the other metrics in this cluster (r = .17).

One possible explanation for seemingly unexpected groupings is methodological rather than conceptual. The *right-branching open dependencies* metric is computed only for dependencies involving a verb, resulting in a value of zero for most words (34% of words in our sample; see also Figure 1). Consequently, its correlations with other sparse syntactic metrics may be inflated by a shared floor effect. For example, *right-branching open dependencies* correlates with *structural distance* (r = .24; Figure 2D), which is another metric that assigns many zero values (58% of words). Excluding observations for which both metrics assign zero reduced this correlation to r = .03, supporting the floor-effect interpretation. Importantly, however, repeating the clustering analysis after excluding shared zero values produced a qualitatively similar clustering pattern, suggesting that the proposed taxonomy is robust to this potential confound.

In addition to the potential influence of shared floor effects, the less consistent clustering observed in the memory dendrogram (Figure 2D) may reflect the proposed sub-taxonomy of memory into anticipation, maintenance, and retrieval. Rather than forming two groups corresponding to the two grammatical formalisms, the memory metrics separated into three main clusters. Although these clusters are quite homogenous in terms of grammatical formalism (two predominantly dependency clusters, one predominantly constituency cluster), they may also correspond to the memory subcomponents of anticipation (upper cluster, predominantly dependency-based metrics), maintenance (middle cluster, predominantly constituency-based metrics), and retrieval (lower cluster, predominantly dependency-based metrics).

To summarize, computation and memory appear to constitute dissociable cognitive components, but only when evaluated for constituency- and dependency-based metrics separately. Likewise, constituency- and dependency-based metrics are largely dissociable when computation and memory are analyzed separately. At the same time, the clustering analysis confirms the idea that many individual metrics do not capture a single cognitive process in isolation, but instead combine multiple aspects of syntactic processing (e.g., the *depth* metrics). Furthermore, similarities among some metrics may be inflated by methodological factors, such as a high proportion of shared zero values. When all metrics are analyzed together, the resulting clusters are thereby less clearly interpretable. Despite this high heterogeneity, the clustering analysis shows

that much of this heterogeneity can be understood in terms of a distinction between grammatical formalisms and cognitive processes. Taken together, these findings provide a principled basis for using the proposed taxonomy as an organizing framework for reviewing the neuroimaging literature on incremental syntactic processing.

5. Neural correlates of syntactic complexity

In this section, we review the brain regions and neural signatures that have been associated with syntactic complexity metrics in neuroimaging research. Our goal is not to provide an exhaustive review of all reported effects, but rather to identify recurring patterns and points of convergence that bear on the cognitive interpretation of these metrics. A comprehensive review would not be feasible, in part because not all metrics have been evaluated against neural data, and in part because only a small subset of studies has directly compared different metrics within the same experiment. A further challenge arises from substantial methodological heterogeneity across studies. Many studies have used fMRI (Bhattasali et al., 2019; Brennan et al., 2012, 2016; Giglio et al., 2024; Li & Hale, 2019; Lopopolo et al., 2021; Shain et al., 2022; Stanojević et al., 2023; Uddén et al., 2022), which offers high spatial resolution and is therefore informative about the cortical distribution of syntactic effects. However, as most of these studies relied on spatially constrained region-of-interest (ROI) analyses, a voxel-wise meta-analysis of syntax-related neural correlates is currently not possible. Other studies have used techniques with high temporal resolution, such as ECoG (Nelson et al., 2017) and MEG (Brennan & Pylkkänen, 2017; Coopmans et al., 2025; Gwilliams et al., 2025; Slaats et al., 2024; Weissbart & Martin, 2024; Zioga et al., 2023), which are better suited for capturing the rapid dynamics of incremental syntactic processing. However, these studies differ widely in the specific neural signals analyzed, varying from oscillatory power in different frequency bands to measures of phase-amplitude coupling. Because these neural markers are thought to index distinct neurocognitive processes, such variability further complicates attempts to compare results across studies and makes it difficult to provide a common cognitive interpretation of existing effects. Given these sources of heterogeneity, we organize the present review around the cognitive distinction between memory-related and computation-related components of syntactic processing. This provides a principled basis for assessing whether different classes of complexity metrics show systematic differences in their neural correlates.

5.1 Disentangling computation and memory

Only a small number of studies have examined syntactic metrics reflecting both memory and computation within a single experiment (Cai et al., 2026; Giglio et al., 2024; Gwilliams et al., 2025; Nelson et al., 2017; Shain et al., 2022; Zioga et al., 2023). While these studies did not explicitly frame their metrics as indexing distinct cognitive components, they could nevertheless be used to evaluate whether memory and computation exhibit dissociable neural signatures.

Two relevant studies relied on constituency-based metrics. In an ECoG experiment, Nelson et al. (2017) used a range of syntactic predictors to model high-gamma power in the left hemisphere during controlled sentence reading. Metrics related to computation (i.e., *bottom-up node count, bottom-up and left-corner parser operations*) and memory (i.e., *open nodes, bottom-up parser stack*) were associated with widely distributed cortical activation patterns. Although the authors did not directly compare the two components, effects linked to memory appear more robust and spatially extensive, particularly in superior temporal regions and in inferior and superior frontal regions. By contrast, computation-related effects were weaker and spatially more restricted, with primary involvement of the superior temporal sulcus (STS) and BA45 of the inferior frontal gyrus (IFG). These findings might suggest that memory places sustained demands on a widely distributed network, whereas computation-related processes may be more transient and focal.

A partially different pattern was reported in an fMRI study by Giglio et al. (2024), which examined spontaneous spoken language comprehension (and production) and restricted analyses to three predefined left-hemispheric ROIs. The memory metric (*open nodes*) was positively associated with activity in the posterior middle temporal gyrus (pMTG) and BA45.[12] However, in contrast to Nelson et al. (2017), the size of the memory effect appeared weaker than the size of the computation effect. Moreover, in comprehension, the computation-related metric (*bottom-up node count*) was confined to posterior temporal regions and did not include the IFG. A similar temporal-dominant pattern for computation has been reported elsewhere (Brennan et al., 2012, 2016; Brennan & Pylkkanen, 2017; Li & Hale, 2019; Lopopolo et al., 2021), though one study has more found frontal involvement for syntactic computation (Bhattasali et al., 2019). Taken together, these

[12] Note that open nodes were not computed identically in Giglio et al. (2024) and Nelson et al. (2017). While both counted the number of nodes that have been opened but not yet closed, opened nodes were counted following an anticipatory top-down strategy in Giglio et al. (2024) but an integrative bottom-up strategy in Nelson et al. (2017).

results suggest that the relative prominence of memory versus computation effects may depend on various factors, including task demands, the nature of the stimulus (isolated sentences, audiobooks, spontaneous speech; see Section 6), and the temporal resolution of the measurement technique.

Evidence from dependency-based metrics does not straightforwardly resolve these discrepancies. In an fMRI study on story listening, Shain et al. (2022) observed strong activation in both inferior frontal and temporal regions of the left-hemispheric language network associated with *DLT integration cost* (a memory-retrieval metric). In contrast, evidence for activity in the language network related to *DLT storage cost* (an anticipatory memory metric) was considerably weaker; it did not explain neural variance above and beyond the effects of *DLT integration cost*. An MEG study involving audiobook listening by Zioga et al. (2023) found significant effects of both memory-related (*remained open dependencies*) and computation-related (*resolved dependencies*) metrics, which modulated alpha- and beta-band power across left-hemispheric regions. Unlike Nelson et al. (2017), however, their memory effects were spatially restricted and primarily left temporal (see also Uddén et al., 2022), whereas the computation effects were stronger overall and encompassed bilateral frontal and temporal areas. It remains unclear how these findings should be reconciled with the above constituency-based results, which suggested substantial frontal involvement for memory.

One potentially promising avenue for understanding these mixed patterns is to consider differences in temporal dynamics rather than anatomical location. Using time-resolved decoding of MEG during audiobook listening, Gwilliams et al. (2025) showed that a memory-related metric (*constituency depth*) was sustained over a relatively long time window, whereas a computation-related component (*bottom-up node count*) exhibited a more transient effect approximately 300 ms after word offset. Such a temporal dissociation suggests that memory and computation may recruit overlapping neural substrates but differ in their duration and temporal engagement (see e.g., Regev et al., 2024). This could help explain why memory effects are more robustly detected across studies, whereas (potentially more transient) computation effects may be more sensitive to methodological differences in temporal resolution.

5.2 Constituency and dependency processing

Only two studies directly compared the neural correlates associated with constituency- and dependency-based metrics. Lopopolo et al. (2021) used fMRI during audiobook listening to

investigate whether different brain regions are differentially related to the kinds of syntactic computations assumed to underlie constituency and dependency structure building. This approach presupposes that constituency and dependency structures are formally distinct and encode non-equivalent aspects of syntactic organization. They observed dissociable responses within the temporal lobe: the dependency-based metric of computation (*resolved dependencies*) was associated with activity in the anterior temporal lobe, whereas the constituency-based metric of computation (*bottom-up node count*) was linked with more posterior regions of the superior temporal cortex. Crucially, these findings do not by themselves warrant the conclusion that anterior and posterior temporal regions support cognitively distinct components of syntactic structure building. Such an interpretation would require an explicit theory specifying which aspects of syntactic computation differ between the two frameworks, and how these differences map onto neural mechanisms (i.e., where exactly these frameworks diverge and make different predictions about processing effort). Rather, the results suggest that constituency- and dependency-based metrics are not interchangeable at the empirical level.

Converging evidence comes from a recent study that recorded single-neuron activity during spontaneous sentence production (Cai et al., 2026). They identified distinct neuronal populations selectively responsive to *constituency depth* and *dependency depth*, two memory metrics. These neurons were broadly distributed across the frontotemporal cortex and did not reveal systematic regional differences between the two metrics. Nevertheless, the fact that the overlap between neuronal populations encoding constituency and dependency depth was limited suggests that even closely related grammatical quantifications of memory index partially distinct aspects of syntactic processing. This again demonstrates that representational assumptions have measurable empirical consequences. Selecting one grammatical framework over another determines which aspects of syntactic structure are quantified and thereby shapes the observed relationship between syntactic complexity and neural activity (see Section 2).

6. Discussion and future directions

Our review reveals substantial heterogeneity in how syntactic complexity is operationalized and interpreted across studies. Existing studies employ a wide range of complexity metrics, reflecting divergent assumptions about the nature of grammatical representations and the cognitive locus of processing effort. Because these assumptions are rarely made explicit—i.e., studies frequently

simply state that they measure “syntactic processing” without specifying the cognitive dimension being quantified—these inconsistencies often go unnoticed. As discussed in Section 5, this complicates the interpretation of reported neural correlates of syntactic processing. To address this issue, we proposed a theoretically motivated taxonomy of existing word-level metrics of syntactic complexity. The taxonomy clarifies the cognitive interpretation of individual metrics, highlights assumptions that are often left implicit, and identifies where ostensibly similar metrics diverge and formally distinct metrics converge.

Although the metrics reviewed here are intended to capture incremental processing, they are typically derived from complete parse trees. When such metrics are applied to human sentence processing, it is assumed that the parser has access to the correct syntactic analysis at every point in the sentence, even in the presence of local ambiguity. This is a useful modeling simplification, but it crucially abstracts away from uncertainty during real-time language comprehension, where multiple structural continuations remain compatible with the input. Human comprehenders must therefore manage uncertainty over competing analyses until disambiguating information becomes available. However, anticipatory metrics like *top-down node count* and *opened dependencies* require specifying how many constituents or dependencies will be projected even before the parser knows how the sentence will continue. As a result, such deterministic tree-based metrics do not capture the probabilistic nature of human sentence processing.

Statistical approaches offer a way to address this limitation. The most prominently used metric is surprisal, which is used to quantify processing difficulty as a function of the probability of an incoming word given the preceding context (see Hale, 2016 for a review). Surprisal is commonly computed over sequences of words or tokens and therefore does not directly capture syntactic properties (Slaats & Martin, 2025). However, not all probabilistic metrics are syntax-agnostic. Probabilistic language models can be used to derive the surprisal of a word’s part of speech, which is assumed to reflect syntactically conditioned expectations about the upcoming word’s syntactic category (e.g., Heilbron et al., 2022; Lopopolo et al., 2017). Here, surprisal is estimated over sequences of part-of-speech tokens, which do not explicitly encode hierarchical structure. Alternative approaches incorporate hierarchical syntactic structure explicitly. For

instance, surprisal can be computed from probabilistic context-free grammars, which assign probabilities to alternative syntactic derivations (e.g., Boston et al., 2008; Brennan et al. 2016; Brennan & Hale, 2019; Demberg & Keller, 2008; Shain et al., 2020; Van Schijndel & Schuler, 2015). Moreover, Recurrent Neural Network Grammars are neural sequence models that jointly model sentences and their constituency trees, allowing them to maintain explicit hierarchical representations while computing probabilistic expectations over future input (Hale et al., 2018; Brennan et al., 2020; Sugimoto et al., 2023). These approaches combine explicit syntactic representations with probabilistic inference over possible structural analyses, complementing the deterministic metrics reviewed here. A limitation of these syntax-aware probabilistic models, however, is their dependence on the availability of trained probabilistic parsers and syntactically annotated corpora, which currently restricts their application across languages.

Several additional avenues for future research deserve attention. First, as is clear from Table 1, only very few—primarily Germanic—languages have been used to assess the predictive validity of different complexity metrics. Expanding the empirical base to include other languages, especially those with grammatical properties different from those of Germanic languages, would be particularly valuable. Such cross-linguistic comparisons should be motivated by explicit hypotheses about which grammatical properties are expected to affect processing, or, alternatively, by the assumption that the cognitive mechanisms under investigation are universal. One interesting point of cross-linguistic variation concerns constraints on word order. Dependency representations are often argued to be better suited to languages with relatively free word order and discontinuous constituents (de Marneffe & Nivre, 2019; Osborne, 2014). If so, dependency-based metrics might be expected to predict processing effort in such languages more accurately than constituency-based metrics. Cross-linguistic comparisons therefore provide an opportunity not merely to replicate existing findings, but to evaluate competing assumptions about the grammatical representations underlying syntactic complexity. Likewise, cross-linguistic variation in head direction provides a way to evaluate competing assumptions about anticipatory versus maintenance-based storage. Whether the head precedes or follows its complements determines when syntactic information becomes available for anticipatory structure building. If structure building depends on access to

syntactic heads, anticipatory metrics should perform differently in head-initial and head-final constructions. For instance, when two otherwise identical structures differ only in head direction, the hierarchical *depth* of preverbal and postverbal complements is the same, but these need not be equally costly to process. For instance, it has been shown that processing measures are sensitive to *DLT storage* in Japanese (a head-final language; Nakatani & Gibson, 2010) but not as much in English (a head-initial language; Shain et al., 2022). Such findings illustrate that expanding cross-linguistic coverage is essential for both evaluating the generality of syntactic complexity metrics and testing the cognitive assumptions on which they are based.

Second, almost all metrics have been evaluated against measures of comprehension, even though some were originally proposed to account for sentence production (e.g., *Yngve score*). Although syntactic representations may be shared between comprehension and production, the information available for incremental processing differs substantially between the two: speakers have access to their intended message and therefore have more information about upcoming syntactic structure than comprehenders, who must infer that structure from the unfolding input (see Momma & Phillips, 2018 for discussion). Consequently, anticipatory metrics may be more suitable for modeling production than comprehension. This prediction is supported by the results of Giglio et al. (2024), who showed that brain activity during sentence production was better predicted by *top-down node count*, whereas sentence comprehension was better accounted for by *bottom-up node count*. These findings suggest that parsing strategies are adapted to the demands of the task, which means that the predictive validity of different metrics depends not only on the grammatical assumptions they embody, but also on whether it is intended to model comprehension or production.

Finally, if parsing strategies adapt to both the language being processed and the task being performed, they may also adapt to the characteristics of the input. Relevant characteristics include whether the input is read or spontaneously produced, and whether it is written or spoken. In contrast to carefully read speech, spontaneously produced speech contains disfluencies, hesitations, and restarts, which create incomplete and sometimes ungrammatical sentence fragments (Agmon et al., 2023, 2024; Giglio et al., 2024). Comprehenders may rely less on

predictive structure building when the input contains fragments and restarts that reduce the reliability of syntactic predictions (e.g., Brothers et al., 2019). Likewise, written and spoken language differ substantially in their syntactic characteristics (Dobrovoljc, 2026). If comprehenders adapt their parsing strategies to these characteristics, effects observed during spoken and written language comprehension may diverge. Understanding these adaptations will help determine under which conditions anticipatory and non-anticipatory metrics provide the best account of human sentence processing.

In sum, naturalistic paradigms have the potential to provide unique insights into how formally defined properties of syntactic structure relate to behavioral and neural measures of language processing. However, this potential has not yet been fully realized, largely because syntactic complexity metrics are often treated as generic measures of "syntactic processing" rather than being interpreted in terms of the specific cognitive processes they are intended to capture. To make progress in linking linguistic structure to cognitive and neural processes, future studies should therefore be more explicit about the theoretical assumptions underlying their choice of complexity metric, and should consider how those assumptions shape the interpretation of their empirical findings. By systematically organizing existing metrics according to their grammatical and cognitive assumptions, this review provides a principled framework for achieving this goal.

References

Abney, S. P., & Johnson, M. (1991). Memory requirements and local ambiguities of parsing strategies. *Journal of Psycholinguistic Research*, *20*(3), 233–250. https://doi.org/10.1007/BF01067217

Agmon, G., Jaeger, M., Tsarfaty, R., Bleichner, M. G., & Zion Golumbic, E. (2023). "Um…, It's Really Difficult to… Um… Speak Fluently": Neural Tracking of Spontaneous Speech. *Neurobiology of Language*, *4*(3), 435–454. https://doi.org/10.1162/nol_a_00109

Agmon, G., Pradhan, S., Ash, S., Nevler, N., Liberman, M., Grossman, M., & Cho, S. (2024). Automated Measures of Syntactic Complexity in Natural Speech Production: Older and Younger Adults as a Case Study. *Journal of Speech, Language, and Hearing Research*, *67*(2), 545–561. https://doi.org/10.1044/2023_JSLHR-23-00009

Ahrenberg, L. (2007). LinES: An English-Swedish Parallel Treebank. In J. Nivre, H.-J. Kaalep, K. Muischnek, & M. Koit (Eds.), *Proceedings of the 16th Nordic Conference of Computational Linguistics (NODALIDA 2007)* (pp. 270–273). University of Tartu, Estonia. https://aclanthology.org/W07-2441/

Baumann, P. (2014). Dependencies and hierarchical structure in sentence processing. *Proceedings of the 36th Annual Meeting of the Cognitive Science Society*, 152–157.

Bhattasali, S., Fabre, M., Luh, W.-M., Saied, H. A., Constant, M., Pallier, C., Brennan, J. R., Spreng, R. N., & Hale, J. (2019). Localising memory retrieval and syntactic composition: An fMRI study of naturalistic language comprehension. *Language, Cognition and Neuroscience*, *34*(4), 491–510. https://doi.org/10.1080/23273798.2018.1518533

Bock, J. K. (1982). Toward a cognitive psychology of syntax: Information processing contributions to sentence formulation. *Psychological Review*, *89*(1), 1–47. https://doi.org/10.1037/0033-295X.89.1.1

Boston, M. F., 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. https://doi.org/10.16910/jemr.2.1.1

Brennan, J. (2016). Naturalistic sentence comprehension in the brain. *Language and Linguistics Compass*, *10*(7), 299–313. https://doi.org/10.1111/lnc3.12198

Brennan, J. R., Dyer, C., Kuncoro, A., & Hale, J. T. (2020). Localizing syntactic predictions using recurrent neural network grammars. *Neuropsychologia*, *146*, 107479. https://doi.org/10.1016/j.neuropsychologia.2020.107479

Brennan, J. R., & Hale, J. T. (2019). Hierarchical structure guides rapid linguistic predictions during naturalistic listening. *PLOS ONE*, *14*(1), e0207741. https://doi.org/10.1371/journal.pone.0207741

Brennan, J. R., Nir, Y., Hasson, U., Malach, R., Heeger, D. J., & Pylkkänen, L. (2012). Syntactic structure building in the anterior temporal lobe during natural story listening. *Brain and Language*, *120*(2), 163–173. https://doi.org/10.1016/j.bandl.2010.04.002

Brennan, J. R., & Pylkkänen, L. (2017). MEG evidence for incremental sentence composition in the anterior temporal lobe. *Cognitive Science*, *41*(S6), 1515–1531. https://doi.org/10.1111/cogs.12445

Brennan, J. R., Stabler, E. P., Van Wagenen, S. E., Luh, W.-M., & Hale, J. T. (2016). Abstract linguistic structure correlates with temporal activity during naturalistic comprehension. *Brain and Language*, *157–158*, 81–94. https://doi.org/10.1016/j.bandl.2016.04.008

Brothers, T., Dave, S., Hoversten, L. J., Traxler, M. J., & Swaab, T. Y. (2019). Flexible predictions during listening comprehension: Speaker reliability affects anticipatory processes. *Neuropsychologia*, *135*, 107225. https://doi.org/10.1016/j.neuropsychologia.2019.107225

Cai, J., Kfir, Y., Jamali, M., Huang, H., Kim, Y. J., Cash, S. S., & Williams, Z. M. (2026). Mapping the neuronal building blocks of human language with language models. *Nature*, 1–9. https://doi.org/10.1038/s41586-026-10691-5

Caplan, D., Vijayan, S., Kuperberg, G., West, C., Waters, G., Greve, D., & Dale, A. M. (2001). Vascular responses to syntactic processing: Event-related fMRI study of relative clauses. *Human Brain Mapping*, *15*(1), 26–38. https://doi.org/10.1002/hbm.1059

Carnie, A. (2021). *Syntax: A generative introduction*. Wiley-Blackwell.

Chomsky, N. (1957). *Syntactic structures*. Mouton.

Chomsky, N. (1972). *Studies on Semantics in Generative Grammar*. Walter de Gruyter.

Collins, C., & Stabler, E. (2016). A Formalization of Minimalist Syntax. *Syntax*, *19*(1), 43–78. https://doi.org/10.1111/synt.12117

Coopmans, C. W., de Hoop, H., Hagoort, P., & Martin, A. E. (2022). Effects of structure and meaning on cortical tracking of linguistic units in naturalistic speech. *Neurobiology of Language*, *3*(3), 386–412. https://doi.org/10.1162/nol_a_00070

Coopmans, C. W., de Hoop, H., Kaushik, K., Hagoort, P., & Martin, A. E. (2022). Hierarchy in language interpretation: Evidence from behavioural experiments and computational modelling. *Language, Cognition and Neuroscience*, *37*(4), 420–439. https://doi.org/10.1080/23273798.2021.1980595

Coopmans, C. W., Hoop, H. de, Tezcan, F., Hagoort, P., & Martin, A. E. (2025). Language-specific neural dynamics extend syntax into the time domain. *PLOS Biology*, *23*(1), e3002968. https://doi.org/10.1371/journal.pbio.3002968

Coopmans, C. W., Kaushik, K., & Martin, A. E. (2023). Hierarchical structure in language and action: A formal comparison. *Psychological Review*, *130*(4), 935–952. https://doi.org/10.1037/rev0000429

Coopmans, C. W., & Zaccarella, E. (2023). Three conceptual clarifications about syntax and the brain. *Frontiers in Language Sciences*, *2*. https://www.frontiersin.org/articles/10.3389/flang.2023.1218123

Dahl, Ö. (1980). Some arguments for higher nodes in syntax: A reply to Hudson's 'Constituency and dependency.' *Linguistics*, *18*(5–6), 485–488. https://doi.org/10.1515/ling.1980.18.5-6.485

de Marneffe, M.-C., Manning, C. D., Nivre, J., & Zeman, D. (2021). Universal Dependencies. *Computational Linguistics*, *47*(2), 255–308. https://doi.org/10.1162/coli_a_00402

de Marneffe, M.-C., & Nivre, J. (2019). Dependency Grammar. *Annual Review of Linguistics*, *5*, 197–218. https://doi.org/10.1146/annurev-linguistics-011718-011842

Dobrovoljc, K. (2026). Counting trees: A treebank-driven exploration of syntactic variation in speech and writing across languages. *Corpus Linguistics and Linguistic Theory*, 1–37. https://doi.org/10.1515/cllt-2025-0046

Embick, D., & Poeppel, D. (2015). Towards a computational(ist) neurobiology of language: Correlational, integrated and explanatory neurolinguistics. *Language, Cognition and Neuroscience*, *30*(4), 357–366. https://doi.org/10.1080/23273798.2014.980750

Ferreira, F., & Engelhardt, P. E. (2006). Syntax and Production. In M. J. Traxler & M. A. Gernsbacher (Eds.), *Handbook of Psycholinguistics (Second Edition)* (pp. 61–91). Academic Press. https://doi.org/10.1016/B978-012369374-7/50004-3

Fodor, J. A., Bever, T. G., & Garrett, M. F. (1974). *The Psychology of Language: An Introduction to Psycholinguistics and Generative Grammar*. McGraw-Hill.

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). MIT Press.

Frazier, L. (1985). Syntactic complexity. In A. M. Zwicky, D. R. Dowty, & L. Karttunen (Eds.), *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives* (pp. 129–189). Cambridge University Press. https://doi.org/10.1017/CBO9780511597855.005

Frazier, L. (1987). Sentence Processing: A Tutorial Review. In M. Coltheart (Ed.), *Attention and Performance XII* (pp. 559-586). Routledge.

Futrell, R., Mahowald, K., & Gibson, E. (2015). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, *112*(33), 10336–10341. https://doi.org/10.1073/pnas.1502134112

Gaifman, H. (1965). Dependency systems and phrase-structure systems. *Information and Control*, *8*(3), 304–337. https://doi.org/10.1016/S0019-9958(65)90232-9

Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, *68*(1), 1–76. https://doi.org/10.1016/S0010-0277(98)00034-1

Gibson, E. (2000). The dependency locality theory: A distance-based theory of linguistic complexity. In A. Marantz, Y. Miyashita, & W. O'Neil (Eds.), *Image, language, brain* (Vol. 2000, pp. 95–126). MIT Press.

Gibson, E. (2025). *Syntax: A cognitive approach.* MIT Press.

Giglio, L., Ostarek, M., Sharoh, D., & Hagoort, P. (2024). Diverging neural dynamics for syntactic structure building in naturalistic speaking and listening. *Proceedings of the National Academy of Sciences*, *121*(11), e2310766121. https://doi.org/10.1073/pnas.2310766121

Gordon, P. C., Hendrick, R., & Johnson, M. (2001). Memory interference during language processing. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, *27*(6), 1411–1423. https://doi.org/10.1037/0278-7393.27.6.1411

Grodner, D., & Gibson, E. (2005). Consequences of the Serial Nature of Linguistic Input for Sentenial Complexity. *Cognitive Science*, *29*(2), 261–290. https://doi.org/10.1207/s15516709cog0000_7

Grodzinsky, Y., & Friederici, A. D. (2006). Neuroimaging of syntax and syntactic processing. *Current Opinion in Neurobiology, Cognitive Neuroscience*, *16*(2), 240–246. https://doi.org/10.1016/j.conb.2006.03.007

Gwilliams, L., Marantz, A., Poeppel, D., & King, J.-R. (2025). Hierarchical dynamic coding coordinates speech comprehension in the human brain. *Proceedings of the National Academy of Sciences*, *122*(42), e2422097122. https://doi.org/10.1073/pnas.2422097122

Hagoort, P., Brown, C. M., & Osterhout, L. (1999). The neurocognition of syntactic processing. In C. M. Brown & P. Hagoort (Eds.), *The neurocognition of language* (pp. 273–317). Oxford University Press.

Hagoort, P., & Indefrey, P. (2014). The neurobiology of language beyond single words. *Annual Review of Neuroscience*, *37*(1), 347–362. https://doi.org/10.1146/annurev-neuro-071013-013847

Hale, J., Dyer, C., Kuncoro, A., & Brennan, J. (2018). Finding syntax in human encephalography with beam search. *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2727–2736. https://doi.org/10.18653/v1/P18-1254

Hale, J. T. (2014). *Automaton theories of human sentence comprehension*. CSLI Publications.

Hale, J. (2016). Information-theoretical Complexity Metrics. *Language and Linguistics Compass*, *10*(9), 397–412. https://doi.org/10.1111/lnc3.12196

Hale, J. T., Campanelli, L., Li, J., Bhattasali, S., Pallier, C., & Brennan, J. R. (2022). Neurocomputational models of language processing. *Annual Review of Linguistics*, *8*(1), 427–446. https://doi.org/10.1146/annurev-linguistics-051421-020803

Hamburger, H., & Crain, S. (1984). Acquisition of cognitive compiling. *Cognition*, *17*(2), 85–136. https://doi.org/10.1016/0010-0277(84)90015-5

Hamilton, L. S., & Huth, A. G. (2020). The revolution will not be controlled: Natural stimuli in speech neuroscience. *Language, Cognition and Neuroscience*, *35*(5), 573–582. https://doi.org/10.1080/23273798.2018.1499946

Heilbron, M., Armeni, K., Schoffelen, J.-M., Hagoort, P., & de Lange, F. P. (2022). A hierarchy of linguistic predictions during natural language comprehension. *Proceedings of the National Academy of Sciences*, *119*(32), e2201968119. https://doi.org/10.1073/pnas.2201968119

Hudson, R. (1990). *English Word Grammar*. Blackwell.

Johnson, K. (2015). Notational Variants and Invariance in Linguistics. *Mind & Language*, *30*(2), 162–186. https://doi.org/10.1111/mila.12076

Johnson-Laird, P. N. (1983). *Mental models*. Harvard University Press.

Kajikawa, K., Yoshida, R., & Oseki, Y. (2024). Dissociating Syntactic Operations via Composition Count. *Proceedings of the Annual Meeting of the Cognitive Science Society*, *46*. https://escholarship.org/uc/item/2bp2m26p

Kaplan, R. M. (1972). Augmented transition networks as psychological models of sentence comprehension. *Artificial Intelligence*, *3*, 77–100. https://doi.org/10.1016/0004-3702(72)90043-4

Kapteijns, B., & Hintz, F. (2021). Comparing predictors of sentence self-paced reading times: Syntactic complexity versus transitional probability metrics. *PLOS ONE*, *16*(7), e0254546. https://doi.org/10.1371/journal.pone.0254546

Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition*, *2*(1), 15–47. https://doi.org/10.1016/0010-0277(72)90028-5

King, J., & Just, M. A. (1991). Individual differences in syntactic processing: The role of working memory. *Journal of Memory and Language*, *30*(5), 580–602. https://doi.org/10.1016/0749-596X(91)90027-H

Klein, D., & Manning, C. (2004). Corpus-Based Induction of Syntactic Structure: Models of Dependency and Constituency. *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, 478–485. https://doi.org/10.3115/1218955.1219016

Li, J., & Hale, J. (2019). Grammatical predictors for fMRI time-courses. In R. C. Berwick & E. P. Stabler (Eds.), *Minimalist parsing* (pp. 159–173). Oxford University Press. https://doi.org/10.1093/oso/9780198795087.003.0007

Liu, H. (2008). Dependency Distance as a Metric of Language Comprehension Difficulty. *Journal of Cognitive Science*, *9*, 159–191. https://doi.org/10.17791/jcs.2008.9.2.159

Liu, H., Xu, C., & Liang, J. (2017). Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of Life Reviews*, *21*, 171–193. https://doi.org/10.1016/j.plrev.2017.03.002

Liu, W., Xiang, M., & Ding, N. (2026). Active use of latent tree-structured sentence representation in humans and large language models. *Nature Human Behaviour*, *10*(2), 303–316. https://doi.org/10.1038/s41562-025-02297-0

Lopopolo, A., Frank, S. L., Bosch, A. van den, & Willems, R. M. (2017). Using stochastic language models (SLM) to map lexical, syntactic, and phonological information processing in the brain. *PLOS ONE*, *12*(5), e0177794. https://doi.org/10.1371/journal.pone.0177794

Lopopolo, A., van den Bosch, A., Petersson, K.-M., & Willems, R. M. (2021). Distinguishing syntactic operations in the brain: Dependency and phrase-structure parsing. *Neurobiology of Language*, *2*(1), 152–175. https://doi.org/10.1162/nol_a_00029

Matchin, W., & Hickok, G. (2020). The cortical organization of syntax. *Cerebral Cortex*, *30*(3), 1481–1498. https://doi.org/10.1093/cercor/bhz180

Marr, D. (1982). *Vision.* W. H. Freeman & Co.

Mel'čuk, I. A. (1988). *Dependency Syntax: Theory and Practice*. State University of New York Press.

Miller, G. A., & Chomsky, N. (1963). Finitary models of language users. In R. D. Luce, R. R. Bush, & E. Galanter (Eds.), *Handbook of mathematical psychology* (Vol. 2, pp. 419–491). Wiley.

Momma, S., & Phillips, C. (2018). The relationship between parsing and generation. *Annual Review of Linguistics*, *4*(1), 233–254. https://doi.org/10.1146/annurev-linguistics-011817-045719

Nakatani, K., & Gibson, E. (2008). Distinguishing theories of syntactic expectation cost in sentence comprehension: Evidence from Japanese. *Linguistics*, *46*(1), 63–87. https://doi.org/10.1515/LING.2008.003

Nastase, S. A., Goldstein, A., & Hasson, U. (2020). Keep it real: Rethinking the primacy of experimental control in cognitive neuroscience. *NeuroImage*, *222*, 117254. https://doi.org/10.1016/j.neuroimage.2020.117254

Nefdt, R. M., & Baggio, G. (2024). Notational Variants and Cognition: The Case of Dependency Grammar. *Erkenntnis*, *89*(7), 2867–2897. https://doi.org/10.1007/s10670-022-00657-0

Nelson, M. J., Karoui, I. E., Giber, K., Yang, X., Cohen, L., Koopman, H., Cash, S. S., Naccache, L., Hale, J. T., Pallier, C., & Dehaene, S. (2017). Neurophysiological dynamics of phrase-structure building during sentence processing. *Proceedings of the National Academy of Sciences*, *114*(18), E3669–E3678. https://doi.org/10.1073/pnas.1701590114

Nivre, J., de Marneffe, M.-C., Ginter, F., Goldberg, Y., Hajič, J., Manning, C. D., McDonald, R., Petrov, S., Pyysalo, S., Silveira, N., Tsarfaty, R., & Zeman, D. (2016). Universal Dependencies v1: A Multilingual Treebank Collection. In N. Calzolari, K. Choukri, T. Declerck, S. Goggi, M.

Grobelnik, B. Maegaard, J. Mariani, H. Mazo, A. Moreno, J. Odijk, & S. Piperidis (Eds.), *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)* (pp. 1659–1666). European Language Resources Association (ELRA). https://aclanthology.org/L16-1262/

Osborne, T. (2014). Dependency Grammar. In A. Carnie, Y. Sato, & D. Siddiqi (Eds.), *The Routledge Handbook of Syntax* (pp. 604–626). Routledge.

Pallier, C., Devauchelle, A.-D., & Dehaene, S. (2011). Cortical representation of the constituent structure of sentences. *Proceedings of the National Academy of Sciences*, *108*(6), 2522–2527. https://doi.org/10.1073/pnas.1018711108

Poeppel, D., & Embick, D. (2005). Defining the relation between linguistics and neuroscience. In A. Cutler (Ed.), *Twenty-first century psycholinguistics: Four cornerstones* (pp. 103–118). Routledge.

Pritchett, B. L. (1992). *Grammatical Competence and Parsing Performance*. University of Chicago Press.

Reddy, A. J., & Wehbe, L. (2021). Can fMRI reveal the representation of syntactic structure in the brain? *Advances in Neural Information Processing Systems*, *34*, 9843–9856.

Regev, T. I., Casto, C., Hosseini, E. A., Adamek, M., Ritaccio, A. L., Willie, J. T., Brunner, P., & Fedorenko, E. (2024). Neural populations in the language network differ in the size of their temporal receptive windows. *Nature Human Behaviour*, *8*(10), 1924–1942. https://doi.org/10.1038/s41562-024-01944-2

Reinhart, T. (1983). Anaphora and semantic interpretation. Croom Helm.

Rezaii, N., Mahowald, K., Ryskin, R., Dickerson, B., & Gibson, E. (2022). A syntax–lexicon trade-off in language production. *Proceedings of the National Academy of Sciences*, *119*(25), e2120203119. https://doi.org/10.1073/pnas.2120203119

Sag, I. A., Wasow, T., & Bender, E. M. (2003). *Syntactic Theory: A Formal Introduction*. CSLI Publications.

Scontras, G., Badecker, W., Shank, L., Lim, E., & Fedorenko, E. (2015). Syntactic Complexity Effects in Sentence Production. *Cognitive Science*, *39*(3), 559–583. https://doi.org/10.1111/cogs.12168

Shain, C., Blank, I. A., Fedorenko, E., Gibson, E., & Schuler, W. (2022). Robust Effects of Working Memory Demand during Naturalistic Language Comprehension in Language-Selective Cortex. *Journal of Neuroscience*, *42*(39), 7412–7430. https://doi.org/10.1523/JNEUROSCI.1894-21.2022

Shain, C., van Schijndel, M., Futrell, R., Gibson, E., & Schuler, W. (2016). Memory access during incremental sentence processing causes reading time latency. In D. Brunato, F. Dell'Orletta, G. Venturi, T. François, & P. Blache (Eds.), *Proceedings of the Workshop on Computational Linguistics for Linguistic Complexity (CL4LC)* (pp. 49–58). The COLING 2016 Organizing Committee. https://aclanthology.org/W16-4106/

Slaats, S., & Martin, A. E. (2025). What's Surprising About Surprisal. *Computational Brain & Behavior*, *8*(2), 233-248. https://doi.org/10.1007/s42113-025-00237-9

Slaats, S., Meyer, A. S., & Martin, A. E. (2024). Lexical Surprisal Shapes the Time Course of Syntactic Structure Building. *Neurobiology of Language*, *5*(4), 942–980. https://doi.org/10.1162/nol_a_00155

Stanojević, M., Brennan, J. R., Dunagan, D., Steedman, M., & Hale, J. T. (2023). Modeling Structure-Building in the Brain With CCG Parsing and Large Language Models. *Cognitive Science*, *47*(7), e13312. https://doi.org/10.1111/cogs.13312

Steedman, M. (2000). *The syntactic process*. MIT press.

Sugimoto, Y., Yoshida, R., Jeong, H., Koizumi, M., Brennan, J. R., & Oseki, Y. (2023). Localizing Syntactic Composition with Left-Corner Recurrent Neural Network Grammars. *Neurobiology of Language*, *5*(1), 201–224. https://doi.org/10.1162/nol_a_00118

Tesnière, L. (1959/2015). *Elements of Structural Syntax*. John Benjamins Publishing Company. https://doi.org/10.1075/z.185

Townsend, D. J., & Bever, T. G. (2001). *Sentence comprehension: The integration of habits and rules*. MIT Press.

Uddén, J., Hultén, A., Schoffelen, J.-M., Lam, N., Harbusch, K., van den Bosch, A., Kempen, G., Petersson, K. M., & Hagoort, P. (2022). Supramodal Sentence Processing in the Human Brain: fMRI Evidence for the Influence of Syntactic Complexity in More Than 200 Participants. *Neurobiology of Language*, *3*(4), 575–598. https://doi.org/10.1162/nol_a_00076

van der Burght, C. L., Friederici, A. D., Maran, M., Papitto, G., Pyatigorskaya, E., Schroën, J. A. M., Trettenbrein, P. C., & Zaccarella, E. (2023). Cleaning up the Brickyard: How Theory and Methodology Shape Experiments in Cognitive Neuroscience of Language. *Journal of Cognitive Neuroscience*, *35*(12), 2067–2088. https://doi.org/10.1162/jocn_a_02058

van Schijndel, M., & Schuler, W. (2015). Hierarchic syntax improves reading time prediction. *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 1597–1605. https://doi.org/10.3115/v1/N15-1183

Weissbart, H., & Martin, A. E. (2024). The structure and statistics of language jointly shape cross-frequency neural dynamics during spoken language comprehension. *Nature Communications*, *15*(1), 8850. https://doi.org/10.1038/s41467-024-53128-1

Yngve, V. H. (1960). A Model and an Hypothesis for Language Structure. *Proceedings of the American Philosophical Society*, *104*(5), 444–466.

Zaccarella, E., Schell, M., & Friederici, A. D. (2017). Reviewing the functional basis of the syntactic Merge mechanism for language: A coordinate-based activation likelihood estimation meta-analysis. *Neuroscience & Biobehavioral Reviews*, *80*, 646–656. https://doi.org/10.1016/j.neubiorev.2017.06.011

Zioga, I., Weissbart, H., Lewis, A. G., Haegens, S., & Martin, A. E. (2023). Naturalistic Spoken Language Comprehension Is Supported by Alpha and Beta Oscillations. *Journal of Neuroscience*, *43*(20), 3718–3732. https://doi.org/10.1523/JNEUROSCI.1500-22.2023