

Modeling word and morpheme order in natural language as an efficient tradeoff of memory and surprisal

Michael Hahn, Judith Degen, Richard Futrell

June 24, 2020

Abstract

Memory limitations are known to constrain language comprehension and production, and have been argued to account for crosslinguistic word order regularities. However, a systematic assessment of the role of memory limitations in language structure has proven elusive, in part because it is hard to extract precise large-scale quantitative generalizations about language from existing mechanistic models of memory use in sentence processing. We provide an architecture-independent information-theoretic formalization of memory limitations which enables a simple calculation of the memory efficiency of languages. Our notion of memory efficiency is based on the idea of a *memory-surprisal tradeoff*: a certain level of average surprisal per word can only be achieved at the cost of storing some amount of information about past context. Based on this notion of memory usage, we advance the *Efficient Tradeoff Hypothesis*: the order of elements in natural language is under pressure to enable favorable memory-surprisal tradeoffs. We derive that languages enable more efficient tradeoffs when they exhibit *information locality*: when predictive information about an element is concentrated in its recent past. We provide empirical evidence from three test domains in support of the Efficient Tradeoff Hypothesis: a reanalysis of a miniature artificial language learning experiment, a large-scale study of word order in corpora of 54 languages, and an analysis of morpheme order in two agglutinative languages. These results suggest that principles of order in natural language can be explained via highly generic cognitively motivated principles and lend support to efficiency-based models of the structure of human language.

1 Introduction

Natural language is a powerful tool that allows humans to communicate, albeit under inherent cognitive resource limitations. Here, we investigate whether human languages are grammatically structured in a way that reduces the cognitive resource requirements for comprehension, compared to counterfactual languages that differ in grammatical structure.

The suggestion that the structure of human language reflects a need for efficient processing under resource limitations has been present in the linguistics and cognitive science literature for decades (Yngve, 1960; Berwick and Weinberg, 1984; Hawkins, 1994; Chomsky, 2005; Jaeger and Tily, 2011; Gibson et al., 2019; Hahn et al., 2020). The idea has been summed up in Hawkins’s (2004) **Performance–Grammar Correspondence Hypothesis** (PGCH), which holds that grammars are structured so that the typical utterance is easy to produce and comprehend under performance constraints.

One major source of resource limitation in language processing is incremental memory use. When producing and comprehending language in real time, a language user must keep track of what they have already produced or heard in some kind of incremental memory store, which is subject to resource constraints. These memory constraints have been argued to underlie various **locality principles** which linguists have used to

predict the orders of words within sentences and morphemes within words (e.g. Behaghel, 1932; Givón, 1985; Bybee, 1985; Rijkhoff, 1990; Hawkins, 1994, 2004, 2014; Temperley and Gildea, 2018). The idea is that language should be structured to reduce long-term dependencies of various kinds, by placing elements that depend on each other close to each other in linear order. That is, elements of utterances which are more ‘relevant’ or ‘mentally connected’ to each other are closer to each other.

Our contribution is to present a new, highly general formalization of the relationship between sequential order and incremental memory in language processing, from which we can derive a precise and empirically testable version of the idea that utterance elements which depend on each other should be close to each other. Our formalization allows us to predict the order of words within sentences, and morphemes within words directly by the minimization of memory usage.

We formalize the notion of memory constraints in terms of what we call the **memory–surprisal tradeoff**: the idea that the ease of comprehension depends on the amount of computational resources invested into remembering previous linguistic elements, e.g., words. Therefore, there exists a tradeoff between the quantity of memory resources invested, and the ease of language processing. The shape of this tradeoff depends on the grammar of a language, and in particular the way that it structures information in time. We characterize memory resources using the theory of lossy data compression (Cover and Thomas, 2006; Berger, 2003), which has recently had success in predicting cognitive phenomena (Sims, 2018; Zénon et al., 2019).

Within our framework, we prove a theorem showing that lower memory requirements result when utterance elements that depend on each other statistically are placed close to each other. This theorem does not require any assumptions about the architecture or functioning of memory, except that it has a bounded capacity. Using this concept, we introduce the **Efficient Tradeoff Hypothesis**: Order in natural language is structured so as to provide efficient memory–surprisal tradeoff curves. We provide evidence for this hypothesis in three studies. We demonstrate that word orders with short dependencies do indeed engender lower working memory resource requirements in toy languages studied in the previous literature, and we show that real word orders in corpora of 54 languages have lower memory requirements than would be expected under artificial baseline comparison grammars. Finally, we show that we can predict the order of morphemes within words in two languages using our principle of the minimization of memory usage.

Our work not only formalizes and tests an old idea in functional linguistics and psycholinguistics, it also opens up connections between those fields and the statistical analysis of natural language (Debowski, 2011; Bentz et al., 2017b; Lin and Tegmark, 2017), and more broadly, between linguistics and fields that have studied information-processing costs and resource requirements in brains (e.g., Friston, 2010) and general physical systems (e.g., Still et al., 2012).

2 Background

A wide range of work has argued that information in natural language utterances is ordered in ways that reduce memory effort, by placing elements close together when they depend on each other in some way. Here, we review these arguments from linguistic and cognitive perspectives.

2.1 Dependency locality and memory constraints in psycholinguistics

When producing and comprehending language in real time, a language user must keep track of what she has already produced or heard in some kind of incremental memory store, which is subject to resource constraints. An early example of this idea is Miller and Chomsky (1963), who attributed the unacceptability of multiple center embeddings in English to limitations of human working memory. Concurrent and subse-

quent work studied how different grammars induce different memory requirements in terms of the number of symbols that must be stored at each point to produce or parse a sentence (Yngve, 1960; Abney and Johnson, 1991; Gibson, 1991; Resnik, 1992). In psycholinguistic studies, memory constraints typically manifest in the form of processing difficulty associated with long-term dependencies. For example, at the level of word-by-word online language comprehension, there is observable processing difficulty at moments when it seems that information about a word must be retrieved from working memory. This difficulty increases when there is a great deal of time or intervening material between the point when a word is first encountered and the point when it must be retrieved from memory (Gibson, 1998; Gibson and Thomas, 1999; Gibson, 2000; McElree, 2000; Lewis and Vasishth, 2005; Bartek et al., 2011; Nicenboim et al., 2015; Balling and Kizach, 2017). That is, language comprehension is harder for humans when words which depend on each other for their meaning are separated by many intervening words. This idea is most prominently associated with the **Dependency Locality Theory** of human sentence processing (Gibson, 2000).

For example, Grodner and Gibson (2005) studied word-by-word reading times in a series of sentences such as (1) below.

- (1) a. The administrator who the nurse supervised...
- b. The administrator who the nurse from the clinic supervised...
- c. The administrator who the nurse who was from the clinic supervised...

In these sentences, the distance between the noun *administrator* and the verb *supervised* is successively increased. Grodner and Gibson (2005) found that as this distance increases, there is a concomitant increase in reading time at the verb *supervised* and following words.

The hypothesized reason for this reading time pattern is based on memory constraints. The idea goes: at the word *supervised*, a comprehender who is trying to compute the meaning of the sentence must integrate a representation of the verb *supervised* with a representation of the noun *administrator*, which is a direct object of the verb. This integration requires retrieving the representation of *administrator* from working memory. If this representation has been in working memory for a long time—for example as in Sentence 1c as opposed to 1a—then the retrieval is difficult or inaccurate, in a way that manifests as increased reading time. Essentially, there exists a dependency between the words *administrator* and *supervised*, and more excess processing difficulty is incurred the more the two words are separated; this excess difficulty is called a **dependency locality effect**.

The existence of dependency locality effects in human language processing, and their connection with working memory, are well-established (Fedorenko et al., 2013). These locality effects in online processing mirror locality effects in word order, described below.

2.2 Locality and cross-linguistic universals of order

Dependency locality in word order means that there is a pressure for words which depend on each other syntactically to be close to each other in linear order. There is ample evidence from corpus statistics indicating that dependency locality is a real property of word order across many languages (Gildea and Temperley, 2007; Liu, 2008; Gildea and Temperley, 2010; Futrell et al., 2015a; Liu et al., 2017; Temperley and Gildea, 2018). Hawkins (1994, 2003) formulates dependency locality as the Principle of Domain Minimization, and has shown that this principle can explain cross-linguistic universals of word order that have been documented by linguistic typologists for decades (Greenberg, 1963). Such a pressure can be motivated in terms of the documented online processing difficulty associated with long-term dependencies among words: dependency locality in word order means that online processing is easier.

An example is order alternation in postverbal constituents in English. While NP objects ordinarily precede PPs (2a, example from Staub et al. (2006)), this order is less preferred when the NP is very long (2c), in which case the inverse order becomes more felicitous (2d). The pattern in (2d) is known as Heavy NP Shift (Ross, 1967; Arnold et al., 2000; Stallings and MacDonald, 2011). Compared to (2c), it reduces the distance between the verb “ate” and the PP, while only modestly increasing the distance between the verb and object NP.

- (2) a. Lucy ate [the broccoli] with a fork.
- b. ? Lucy ate with a fork [the broccoli].
- c. Lucy ate [the extremely delicious, bright green broccoli] with a fork .
- d. Lucy ate with a fork [the extremely delicious, bright green broccoli].

Locality principles have also appeared in a more general form in the functional linguistics literature, in the form of the idea that elements which are more ‘relevant’ to each other will appear closer to each other in linear order in utterances (Behaghel, 1932; Givón, 1985; Givón, 1991; Bybee, 1985; Newmeyer, 1992). Here, ‘elements’ can refer to words or morphemes, and the definition of ‘relevance’ varies. For example, Givón (1985)’s **Proximity Principle** states that elements are placed closer together in a sentence if they are closer conceptually. Applying a similar principle, Bybee (1985) studied the order of morphemes within words across languages, and argued that (for example) morphemes that indicate the valence of a verb (whether it takes zero, one, or two objects) are placed closer to the verb root than morphemes that indicate the plurality of the subject of the verb, because the valence morphemes are more ‘relevant’ to the verb root.

While these theories are widespread in the linguistics literature, there exists to date no quantifiable definition of ‘relevance’ or ‘being closer conceptually’. One of our contributions is to derive such a notion of ‘relevance’ from the minimization of memory usage during language processing.

2.3 Architectural assumptions

The connection between memory resources and locality principles relies on the idea that limitations in working memory will give rise to difficulty when elements that depend on each other are separated at a large distance in time. In previous work, this idea has been motivated in terms of specific assumptions about the architecture of memory. For example, models of memory in sentence processing differ in whether they assume limitations in storage capacity (e.g., ‘memory cost’ in the model of Gibson, 1998) or the precision with which specific elements can be retrieved from memory (e.g. Lewis and Vasishth, 2005). Furthermore, in order to derive the connection between memory usage in such models and locality in word order, it has been necessary to stipulate that memory representations or activations decay over time in some way to explain why longer dependencies are harder to process. The question remains of whether these assumptions about memory architecture are necessary, or whether word orders across languages are optimized for memory independently of the implementation and architecture of human language processing.

In this work, we adopt an information-theoretic perspective on memory use in language processing, which abstracts away from the details of memory architecture. Within our framework, we will establish the connection between memory resources and locality principles by providing general information-theoretic lower bounds on memory use. Our result immediately entails a link between locality and boundedness of memory, following only from the stipulation that memory has bounded capacity measured in bits. In particular, our model does not require any assumption that memory representations or activations decay

over time (as was required in Gibson, 1998; Lewis and Vasishth, 2005; Futrell et al., 2020). We will then show empirical evidence that the orders of words and morphemes in natural language are structured in a way that reduces our measure of memory use compared to the orders of counterfactual baseline languages.

The remainder of the paper is structured as follows. We first introduce the memory-surprisal tradeoff, show that information locality is a property of languages with more efficient memory-surprisal tradeoffs, and introduce the main hypothesis that emerges from the account: the Efficient Tradeoff Hypothesis. Then, we test the Efficient Tradeoff Hypothesis. In Study 1, we qualitatively test the Hypothesis in a reanalysis of word orders emerging in a miniature artificial language study (Fedzechkina et al., 2017). In Study 2, we quantitatively test the Hypothesis in a large-scale study of the word order of 54 languages. In Study 3, we test the Hypothesis on morpheme order in Japanese and Sesotho. Finally, we discuss the implications and limitations of the reported results.

3 Memory-Surprisal Tradeoff

In this section, we introduce the main concept and hypothesis of the paper. We provide a technical definition of the memory-surprisal tradeoff curve, and we prove a theorem showing that more efficient memory-surprisal tradeoffs are possible in languages exhibiting information locality, i.e., in languages where words that depend on each other are close to each other. This theorem establishes the formal link between memory efficiency in online processing and locality in word order.

The memory-surprisal tradeoff applies in both language production and comprehension—more generally, it applies to any system that incrementally produces sequences or extracts information from them. Below, we will develop the theory first in terms of language comprehension; we return to the production perspective in the General Discussion.

3.1 An information-theoretic model of online language comprehension

We begin developing our model by considering the process of language comprehension, where a listener is processing a stream of words uttered by an interlocutor. Experimental research has established three properties of online language comprehension: (1) listeners maintain some information about the words received so far in incremental memory, (2) listeners form probabilistic expectations about the upcoming words (e.g. Altmann and Kamide, 1999; Staub and Clifton Jr, 2006; Kuperberg and Jaeger, 2016), and (3) words are easy to process to the extent that they are predictable based on a listener’s memory of words received so far (Hale, 2001; Levy, 2008; Futrell et al., 2020). See General Discussion for discussion of how our model is related to theories that do not explicitly make these assumptions.

We formalize these three observations into postulates intended to provide a simplified picture of what is known about online language comprehension. Consider a listener comprehending a sequence of words $w_1, \dots, w_t, \dots, w_n$, at an arbitrary time t .

1. Comprehension Postulate 1 (Incremental memory). At time t , the listener has an incremental **memory state** m_t that contains her stored information about previous words. The memory state is characterized by a **memory encoding function** M such that $m_t = M(w_{t-1}, m_{t-1})$.
2. Comprehension Postulate 2 (Incremental prediction). The listener has a subjective probability distribution at time t over the next word w_t as a function of the memory state m_t . This probability distribution is denoted $P(w_t|m_t)$.

3. Comprehension Postulate 3 (Linking hypothesis). Processing a word w_t incurs difficulty proportional to the **surprisal** of w_t given the memory state m_t :

$$\text{Difficulty} \propto -\log P(w_t|m_t). \quad (1)$$

The claim that processing difficulty should be directly proportional to surprisal comes from **surprisal theory** (Hale, 2001; Levy, 2008), an established psycholinguistic theory that can capture reading time effects related to garden-path disambiguation, antilocality effects, and effects of syntactic construction frequency. Surprisal is a robust linear predictor of reading times in large-scale eye-tracking studies based on naturalistic text (Smith and Levy, 2013; Goodkind and Bicknell, 2018), and effects of surprisal have been observed for units as small as phonemes (Gwilliams et al., 2020). Surprisal theory is compatible with different views on the mechanisms underlying prediction, and can reflect different mechanisms such as preactivation and integration (Kuperberg and Jaeger, 2016). We do not assume that listeners explicitly compute a full-fledged distribution $P(w_t|m_t)$; we view $P(w_t|m_t)$ as a formalization of the probabilistic expectations that listeners form during comprehension.

Our expression (1) differs from the usual formulation of surprisal theory in that we consider predictability based on a (potentially lossy or noisy) memory representation m_t , rather than predictability based on the true complete context w_1, \dots, w_{t-1} . The generalization to lossy memory representations is necessary to capture empirically observed effects of memory limitations on language processing, such as dependency locality and structural forgetting (Futrell et al., 2020).

In this work, we are interested in using theories of processing difficulty to derive predictions about languages as a whole, not about individual words or sentences. Therefore, we need a measure of the processing difficulty associated with a language as a whole. For this, we consider the *average* surprisal per word in the language. We call this quantity the **average surprisal** of a language given a memory encoding function M , denoted S_M .

Crucially, the listener’s ability to predict upcoming words accurately depends on how much she remembers about previous words. As the precision of her memory increases, the accuracy of her predictions also increases, and the average surprisal S_M for each incoming word decreases. Taking an information-theoretic perspective, we can think about the amount of information (measured in bits) about previous words stored in the listener’s memory state. This quantity of information is given by the **entropy** of the memory state, which we denote H_M . As the listener stores more and more bits of information about the previous words her memory state, she can achieve lower and lower surprisal values for the upcoming words. This tradeoff between memory and surprisal is the main object of study in this paper.

The **memory–surprisal tradeoff curve** answers the question: for a given amount of information about previous words H_M stored in the listener’s memory state, what is the lowest achievable average surprisal S_M ? Two example tradeoff curves are shown in Figure 1. In general, as the listener stores more information about previous words in her memory state, her lowest achievable average surprisal can only decrease. So the curve is always monotonically decreasing. However, the precise shape of the tradeoff curve depends on the structure of the language being predicted. For example, Figure 1 shows how two hypothetical languages might engender different tradeoff curves, with Language A allowing more favorable tradeoffs than Language B. That is, for Language A, it is possible to achieve lower processing difficulty while investing less memory resources than in Language B.

3.2 Main hypothesis

Having conceptually introduced the memory–surprisal tradeoff, we can state the main hypothesis of this work, the Efficient Tradeoff Hypothesis.

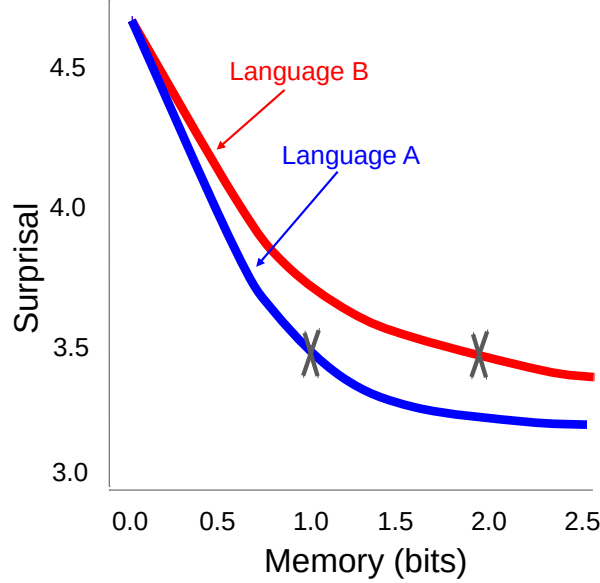


Figure 1: Example memory–surprisal tradeoff curves for two languages, *A* and *B*. Achieving an average surprisal of 3.5 bits requires storing at least 1.0 bits in language *A*, while it requires storing 2.0 bits in language *B*. Language *A* has a steeper memory–surprisal tradeoff than Language *B*, and requires less memory resources to achieve the same level of processing difficulty.

Efficient Tradeoff Hypothesis:

The order of elements in natural language is characterized by a distinctively steeper memory–surprisal tradeoff curve, compared to other possible orders.

A steep tradeoff curve corresponds to memory efficiency, in the sense that it is possible to achieve a low level of processing difficulty (average surprisal S_M) while storing a relatively small amount of information about previous words (entropy of memory H_M). We hypothesize that this property is reflected in grammatical structure and usage preferences across languages.

3.3 Formal definition of the memory–surprisal tradeoff

Here we provide the technical definition of the memory–surprisal tradeoff curve. Let W be a stochastic process generating a stream of symbols extending indefinitely into the past and future: $\dots, w_{-2}, w_{-1}, w_0, w_1, w_2, \dots$. These symbols can represent words, morphemes, or other units for decomposing sentences into a sequence of symbols. We model this process as stationary (Doob, 1953), that is, the joint probability distributions of symbols at different time points depend only on their relative positions in time, not their absolute positions (see SI Section 1.1.1 for more on this modeling assumption).

Let M be a memory encoding function. We consider memory and surprisal costs at an arbitrary time point t . Recall that the surprisal for a specific word w_t after a past word sequence \dots, w_{t-2}, w_{t-1} encoded into a memory state m_t is:

$$-\log P(w_t | m_t). \quad (2)$$

The **average surprisal** of the process W under the memory encoding function M is obtained by averaging over all possible past sequences \dots, w_{t-2}, w_{t-1} with associated memory states m_t , and possible next words

w_t :

$$S_M \equiv \sum_{w_t, m_t} P(m_t) P(w_t | m_t) (-\log P(w_t | m_t)). \quad (3)$$

where w_t ranges over possible symbols, and m_t ranges over possible outputs of the memory encoding function M . This quantity is known as the conditional entropy of w_t given m_t (Cover and Thomas, 2006, p. 17):

$$S_M = H[w_t | m_t]. \quad (4)$$

Because the process W is stationary, the average surprisal S_M is independent of the choice of t (see SI Section 1.1.2). The lowest possible average surprisal for W is attained when m_t perfectly encodes all previous observed words. This quantity is called the **entropy rate** of W (Cover and Thomas, 2006, pp. 74–75):

$$S_\infty \equiv H[w_t | \dots, w_{t-2}, w_{t-1}], \quad (5)$$

which again is independent of t because W is stationary. We use the notation S_∞ to suggest this idea of unlimited resources. The entropy rate of a stochastic process is the irreducible unpredictability of the process: the extent to which a stream of symbols remains unpredictable even for a predictor with unlimited resources. The entropy rate S_∞ of natural language text has been studied by Shannon (1951), Bentz et al. (2017a), and Takahashi and Tanaka-Ishii (2018). Because the memory state m_t is a function of the previous words \dots, w_{t-2}, w_{t-1} , we can prove by the Data Processing Inequality (Cover and Thomas, 2006, pp. 34–35) that the entropy rate must be less than or equal to the average surprisal for any memory encoding function M :

$$S_\infty \leq S_M. \quad (6)$$

If the memory state m_t stores all information about the previous words \dots, w_{t-2}, w_{t-1} , then we have $S_M = S_\infty$.

Having defined average surprisal, we now turn to the question of how to define memory capacity. The average amount of information stored in the memory states m_t is the average number of bits required to encode m_t . This is given by the entropy of the distribution over memory states, H_M :

$$H_M \equiv H[m_t] \quad (7)$$

where

$$H[m_t] = - \sum_m p(m_t = m) \log p(m_t = m) \quad (8)$$

where m runs over all possible states of the memory encoding m_t . Again, because W is stationary, this quantity does not depend on the choice of t (see SI Section 1.1.2).

We will be imposing bounds on H_M and studying the resulting values of S_M .

Definition 1. *The **memory–surprisal tradeoff curve** for a process W is the lowest achievable average surprisal S_M for each value of H_M . Let R denote an upper bound on the memory entropy H_M ; then the memory–surprisal tradeoff curve as a function of R is given by*

$$D(R) \equiv \min_{M: H_M \leq R} S_M, \quad (9)$$

where the minimization is over all memory encoding functions M whose entropy H_M is less than or equal to R .

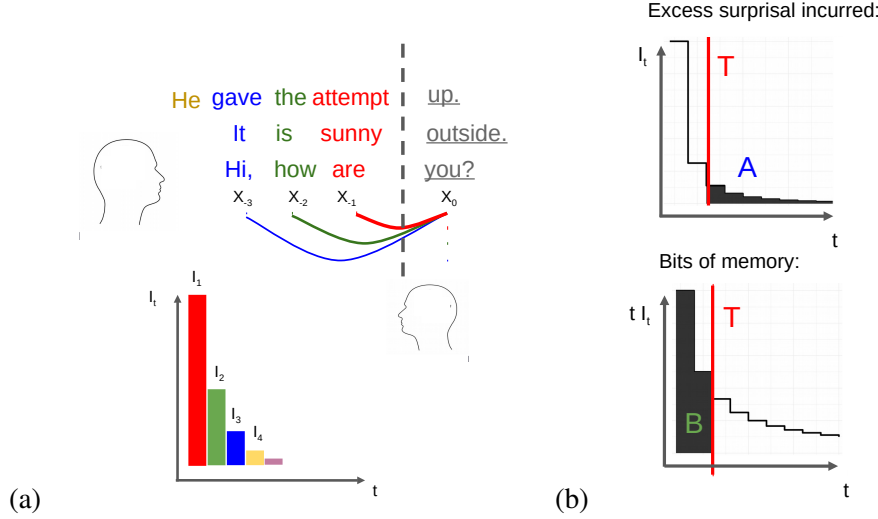


Figure 2: (a) Conditional mutual information I_t captures how much predictive information about the next word is provided, on average, by the word t steps in the past. (b) Here we illustrate our theoretical result. We plot I_t (top) and tI_t (bottom) as functions of t . For any choice of T , a listener using B bits memory (bottom) to represent prior observations will incur at least A bits of extra surprisal beyond the entropy rate (top).

The memory state m_t is generally a lossy representation of the true context of words w_1, \dots, w_{t-1} , meaning that m_t does not contain all the possible information about w_1, \dots, w_{t-1} . The mathematical theory of lossy representations is **rate-distortion theory** (for an overview and key results, see Cover and Thomas, 2006, pp. 301–347); this theory has seen recent successful application in cognitive science and linguistics as a model of rational action under resource constraints (Brady et al., 2009; Sims et al., 2012; Sims, 2018; Zaslavsky et al., 2018; Schach et al., 2018; Zénon et al., 2019; Gershman, 2020). Rate-distortion theory studies curves of the form of Eq. 9, which quantify tradeoffs between negative utility (‘distortion’) and information (‘rate’).

3.4 Information locality

The shape of the memory-surprisal tradeoff is determined in part by the grammatical structure of a language. Some languages enable more efficient tradeoffs than others by allowing a listener to store fewer bits in memory to achieve the same level of average surprisal.

Here, we will demonstrate that the memory-surprisal tradeoff is optimized by languages with word orders exhibiting a property called **information locality**. Information locality means that words that depend on each other statistically are located close to each other in time. We will argue that information locality generalizes the well-known word order principle of dependency locality.

We will make our argument by defining a lower bound on the memory-surprisal tradeoff curve (Eq. 9). This lower bound represents an unavoidable cost associated with a certain level of memory usage H_M ; the true average surprisal S_M might be higher than this bound.

Our argument will make use of a quantity called **mutual information**. Mutual information is the most general measure of statistical association between two random variables. The mutual information between

two random variables X and Y , conditional on a third random variable Z , is defined as:

$$\begin{aligned} I[X : Y|Z] &\equiv \sum_{x,y,z} P(x,y,z) \log \frac{P(x,y|z)}{P(x|z)P(y|z)} \\ &= H[X|Z] - H[X|Y,Z] \\ &= H[Y|Z] - H[Y|X,Z]. \end{aligned} \quad (10)$$

Mutual information is always non-negative. It is zero when X and Y are conditionally independent given Z , and positive whenever X gives any information that makes the value of Y more predictable, or vice versa.

We will study the mutual information structure of natural language sentences, and in particular the mutual information between words at certain distances in linear order. We define the notation I_t to mean the mutual information between words at distance t from each other, conditional on the intervening words:

$$\begin{aligned} I_t &\equiv I[w_t : w_0 | w_1, \dots, w_{t-1}] \\ &= H[w_t | w_1, \dots, w_{t-1}] - H[w_t | w_0, \dots, w_{t-1}]. \end{aligned}$$

This quantity, visualized in Figure 2(a), measures how much predictive information is provided about the current word by the word t steps in the past. It is a statistical property of the language, and can be estimated from large-scale text data.

Equipped with this notion of mutual information at a distance, we can now state our theorem:

Theorem 2. (*Information locality bound*) *For any positive integer T , let M be a memory encoding function such that*

$$H_M \leq \sum_{t=1}^T t I_t. \quad (11)$$

Then we have a lower bound on the average surprisal under the memory encoding function M :

$$S_M \geq S_\infty + \sum_{t=T+1}^{\infty} I_t. \quad (12)$$

A formal proof based on the Comprehension Postulates 1–3 is given in SI Section 1.2.

Interpretation The theorem means that a predictor with limited memory capacity will always be affected by surprisal cost arising from long-term statistical dependencies of length greater than T , for some finite T . This is why we call the result ‘information locality’: processes are easier to predict when most statistical dependencies are short-term (shorter than T). Below we explain in more detail how this interpretation matches the mathematics of the theorem.

The quantities in the theorem are illustrated visually in Figure 2. Eq. 11 describes a memory encoding function which has enough capacity to remember the relevant information from at most T words in the immediate past. The minimal amount of memory capacity which would be required to retain this information is the sum $\sum_{t=1}^T t I_t$, reflecting the cost of holding I_t bits in memory for t timesteps up to $t = T$.

The information locality bound theorem says that the surprisal cost for this memory encoding function is at least $S_\infty + \sum_{t=T+1}^{\infty} I_t$ (Eq. 12). The first term S_∞ is the entropy rate of the process, representing the bits of information in the process which could not have been predicted given any amount of memory. The second term $\sum_{t=T+1}^{\infty} I_t$ is the sum of all the relevant information contained in words *more* than T timesteps in the past (see Figure 2(b)). These correspond to bits of information in the process which *could have* been predicted given infinite memory resources, but which were not, due to the limit on memory usage.

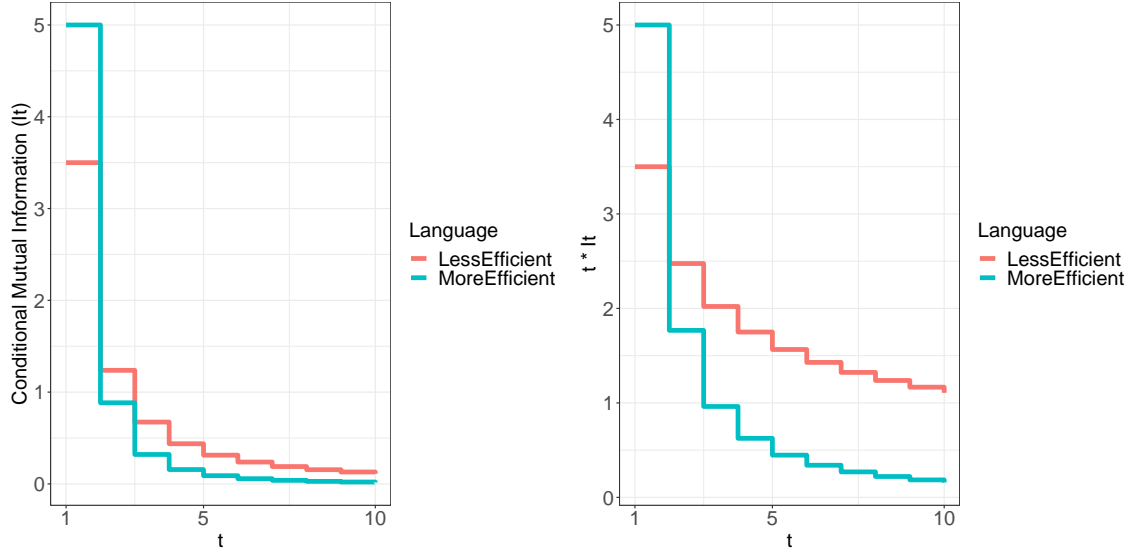


Figure 3: Left: I_t as a function of t , for two different hypothetical languages. I_t decays faster for the MoreEfficient language: Predictive information about the present observation is concentrated more strongly in the recent past. Right: $t \cdot I_t$ as a function of t for the same languages.

The theorem gives a lower bound on the memory–surprisal tradeoff curve, meaning that there is no memory encoding function M with capacity H_M which achieves lower average surprisal than Eq. 12. In terms of psycholinguistics, if memory usage is bounded by Eq. 11, then processing cost of at least Eq. 12 is inevitable. Importantly, the bound holds for *any* memory encoding function M , including functions that do not specifically keep track of a window of the past T words.

Eq. 11 means that maintaining long-term dependencies requires higher memory usage. Carrying the same amount of information over longer distances requires more memory; this fact is reflected in the factor t inside each term of the sum. The result is that modeling long-term statistical dependencies is more costly in terms of memory usage than modeling shorter ones; this cost can manifest in memory storage or in surprisal. The information locality bound theorem demonstrates in a highly general way that language comprehension requires less memory resources when statistical dependencies are mostly short-term.

Because processing long-term dependencies requires higher memory usage, the theorem also implies that a language will be easier to process when most of the predictive information about a word is concentrated close to that word in time—that is, when I_t falls off rapidly as $t \rightarrow \infty$. When memory capacity is limited, then there must be some timescale T such that a listener appears to be affected by excess surprisal arising from statistical dependencies of length greater than T . A language avoids such cost to the extent that it avoids dependencies with a time-span larger than T .

We illustrate the theorem in Figure 3. We consider two hypothetical languages, LessEfficient and MoreEfficient, where $I_t := 5t^{-1.5}$ for LessEfficient and $I_t := 3.5t^{-2.5}$ for MoreEfficient. The curves of I_t , as a function of the distance t , are shown in Figure 3 (left). In both cases, I_t converges to zero as t grows to infinity. However, I_t decays more quickly for language MoreEfficient. This means that predictive information about an observation is concentrated more strongly in the recent past. In Figure 3 (right), we show $t \cdot I_t$ as a function of t . Note that the area under the curve is equal to (11). This area is smaller for the MoreEfficient language, as I_t decays more quickly there. In Figure 4, we show the resulting bounds on memory–surprisal tradeoffs of

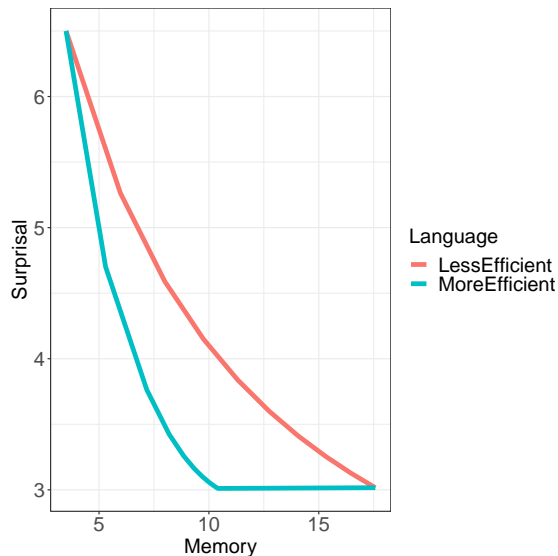


Figure 4: Listener’s memory–surprisal tradeoff for the two hypothetical languages in Figure 3. Recall that the MoreEfficient language has a faster decay of conditional mutual information I_t . Correspondingly, this figure shows that a listener can achieve lower average surprisal at the same level of memory load.

the two languages. As I_t decays faster for language MoreEfficient, it has a more efficient memory–surprisal tradeoff, allowing a listener to achieve strictly lower surprisal across a range of memory values.

4 Study 1: Memory and Dependency Length

So far, we have proven that there exists a tradeoff between memory invested and surprisal incurred during language processing, and that this tradeoff is optimized when languages have relatively short-term dependencies. In this section we qualitatively test the Efficient Tradeoff Hypothesis by reanalyzing the data from Fedzechkina et al. (2017). This is a miniature artificial language study that showed a bias for dependency locality in production in artificial language learning. We will show that, as predicted, the languages which were favored in the artificial language learning experiment are those which optimize the memory–surprisal tradeoff.

4.1 Background: Fedzechkina et al. (2017)

Fedzechkina et al. (2017) conducted a miniature artificial language learning experiment in which participants were exposed to videos describing simple events, paired with sentences in an artificial language of the form Subject–Object–Verb or Object–Subject–Verb, in equal proportion, with free variation between these two word orders. The subject and the object were either simple nouns, or complex noun phrases with modifiers. Participants were trained to produce sentences in response to videos.

Crucially, Fedzechkina et al. (2017) set up the experiment such that in all training trials, either the subject and the object were both simple, or they were both complex. Then, after participants were sufficiently skilled in the use of the artificial language, they were asked to produce sentences describing videos with mixed complexity of noun phrases. The possible word orders that could be produced in this mixed-complexity

A Orders: Short Dependencies

OSV: [[Adjective Noun Postposition] Noun-CASE] Noun Verb

SOV: [[Adjective Noun Postposition] Noun] Noun-CASE Verb

B Orders: Long Dependencies

SOV: Noun [[Adjective Noun Postposition] Noun-CASE] Verb

OSV: Noun-CASE [[Adjective Noun Postposition] Noun] Verb

Figure 5: Production targets in the miniature artificial language from Fedzechkina et al. (2017). The language has head-final order, with free variation between SO and OS orders. When one of the arguments is much longer than the other, placing the longer one first (*A* orders) shortens syntactic dependencies, compared to *B* orders.

setting are shown in Figure 5; the orders marked *A* would create short dependencies, and the orders marked *B* would create long dependencies.

Fedzechkina et al. (2017) found that participants favored the *A* orders over the *B* orders, despite the fact that there was no pattern in the participants’ training input which would have favored *A* over *B*. That is, when exposed to input which was ambiguous with respect to language *A* or *B*, participants favored language *A*. Fedzechkina et al. (2017) explained this result in terms of dependency locality: because the *A* orders create short dependencies between the verb and its arguments and the *B* orders create long dependencies, participants preferred the *A* orders.

4.2 Calculating the memory–surprisal tradeoff for the artificial languages

The Efficient Tradeoff Hypothesis predicts that the favored language *A* has a steeper memory–surprisal tradeoff curve than the disfavored language *B*. Because of the controlled nature of this artificial language, we are able to test this hypothesis by exactly computing the bound on memory as given in Theorem 2. In fact, for this toy process, we can prove that the bound provided by the theorem is achievable, meaning that our computations reflect the true memory–surprisal tradeoff curve, and not only a lower bound on it.

We only consider the head-final version of Fedzechkina et al. (2017)’s artificial language. This is because our bound on the memory–surprisal tradeoff curve is invariant under reversal of a language. That is, if we take a language and reverse the order of all the words in all its sentences, we would measure the same lower bound on the memory–surprisal tradeoff curve (for a proof, see SI Section 1.5). Therefore, strictly head-final and strictly head-initial languages are equivalent under our bound.

We constructed a stochastic process representing the language consisting of sentences with the *A* orders from Figure 5, and one language consisting of the *B* orders. Following the experimental setup of Fedzechkina et al. (2017), we assigned equal probability to the two possible configurations per language, and used a separate set of nouns (inanimate nouns) for the embedded noun in the long phrase.

We interpreted each of the two languages as a stationary processes, extending infinitely in both directions, by concatenating independent samples drawn from the language, and separating them with a special symbol indicating sentence boundaries. We computed the bounds on memory and surprisal (11-12) from Theorem 2 from a chain of 1000 independently sampled sentences, for each of the two versions of the toy language.

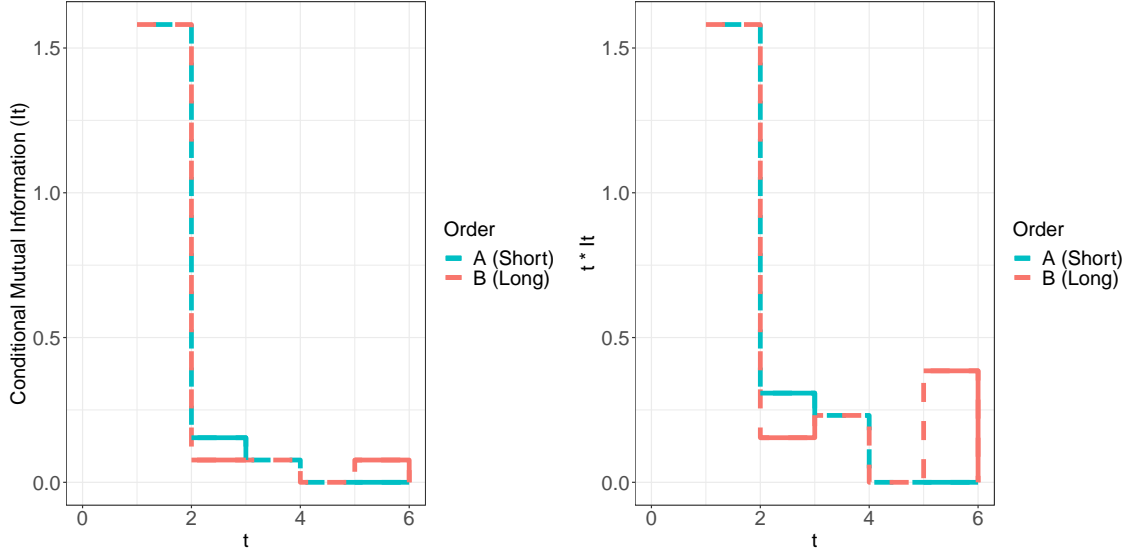


Figure 6: Left: Decay of conditional mutual information I_t , as a function of the distance t , for the two versions in the artificial language. The areas under the two curves are identical, corresponding to the fact that both orders are equally predictable. However, mutual information decays faster in language A. Right: The minimal memory requirement tI_t to store I_t bits of information for timespan t , as a function of t . The area under the B curve is larger, corresponding to larger memory demand for this order.

4.3 Results

Figure 6 (left) shows the curve of the conditional mutual information I_t as a function of the distance t , for the two languages A and B . The curves differ at $t = 2$ and $t = 5$: About 0.073 units of predictive information that are at distance $t = 2$ in the A orders are moved to $t = 5$ in the B orders. The source of the difference lies in predicting the presence and absence of a case marker on the second argument—i.e., whether to anticipate a subject or object. In the A orders, considering the last two words is sufficient to make this decision. In the B orders, it is necessary to consider the word before the long second constituent, which is five words in the past.

The total amount of predictive information—corresponding to the area under the I_t curve—is the same for both languages, indicating that both languages are equally predictable. However, the memory demands differ between the two languages. Figure 6 (right) shows the minimal memory requirements for remembering predictive information at a distance t ($t \cdot I_t$) as a function of t . As I_t decays faster in A orders, the total area under the curve differs between A and B , and is larger in B . Thus, achieving the same predictive accuracy in language B requires more memory resources than in language A .

Figure 7 shows the resulting memory–surprisal tradeoff curve for the two versions of the artificial language from Fedzechkina et al. (2017), obtained by tracing out all values of $T = 1, 2, \dots$ in the theorem, and connecting the points linearly.¹ The curve shows that, at any desired level of surprisal, language A requires at most as much memory as language B . To reach optimal surprisal, the empirically-favored language A requires strictly less memory.

¹Linear interpolation is justified because rate-distortion curves such as the memory–surprisal tradeoff curve are convex (Berger, 2003).

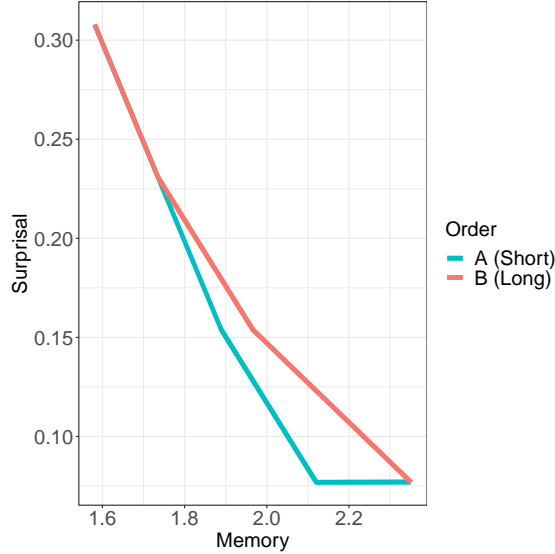


Figure 7: Tradeoff between listener memory and surprisal for the two versions of the artificial language from Fedzechkina et al. (2017). Language A requires less memory at the same level of surprisal.

4.4 Discussion

In a reinterpretation of previous experimental findings, we showed that the languages which are favored in an artificial language learning experiment are those which optimize the memory–surprisal tradeoff. This is evidence that learners and/or speakers have a bias toward word orders that optimize the tradeoff. Furthermore, this result solidifies the link between the memory–surprisal tradeoff and more traditional notions from linguistics, such as dependency locality. We found that the word orders which are optimal from the perspective of dependency locality are also those orders which are optimal from the perspective of the memory–surprisal tradeoff, in the setting of a small controlled artificial language. In Study 2, we scale this approach up to larger corpora of real text.

5 Study 2: Large-Scale Evidence that Word Orders Optimize Memory-Surprisal Tradeoff

To test whether word orders as found in natural language reflect optimization for the memory–surprisal tradeoff more generally, we compare the memory–surprisal tradeoffs of 54 actual languages to those of counterfactual baseline languages. These baseline languages differ from the actual languages only in their word order rules. This method of comparison against counterfactual baseline languages was introduced by Gildea and Temperley (2007, 2010) and has since been fruitfully applied to study optimization-based models of word order universals (Futrell et al., 2015a; Gildea and Jaeger, 2015; Hahn et al., 2020).

Here, we describe how we measure the memory–surprisal tradeoff in corpora, and how we generate counterfactual baseline languages. We then compare the tradeoff in real corpora against the tradeoff in the counterfactual baselines. For the majority of languages, we find that the real languages have more favorable memory–surprisal tradeoffs than the baselines, in line with the Efficient Tradeoff Hypothesis.

5.1 Measuring the memory–surprisal tradeoff in corpora

The key to evaluating the memory–surprisal tradeoff from corpus data is the set of quantities I_t , the mutual information between words at distance t conditional on the intervening words. These quantities can be plugged in to Theorem 2 to give a lower bound on the memory–surprisal tradeoff.

The quantities I_t can be estimated as the difference between the average surprisal of Markov models that have access to the last $t - 1$ or t words. That is, if we have a t th-order Markov model with average surprisal

$$S_t = H[w_t | w_0, \dots, w_{t-1}]$$

and a $(t - 1)$ th-order Markov model with average surprisal

$$S_{t-1} = H[w_t | w_1, \dots, w_{t-1}],$$

then we can calculate I_t straightforwardly in the following way:

$$\begin{aligned} I_t &= I[w_t : w_0 | w_1, \dots, w_{t-1}] \\ &= S_{t-1} - S_t. \end{aligned}$$

Therefore, to evaluate I_t , all we need is a way of fitting Markov models of order t and $t - 1$ and computing their average surprisals.

To fit Markov models to the data, we use neural language models. In particular, we use Recurrent Neural Networks with Long Short-Term Memory architectures (Hochreiter and Schmidhuber, 1997). Neural network models are the basis of the state-of-the-art in statistical modeling of language. Surprisal estimates derived from such models have been shown to best predict reading times, compared to other models, e.g., n -gram models (Frank and Bod, 2011; Goodkind and Bicknell, 2018). See SI Section 3.2 for details on how these models were fit to data, and see SI Sections 3.4 and 3.5 for control studies using other methods of estimating I_t (based on n -gram models and PCFG chart parsers). These control studies yield the same qualitative results as the neural network-based studies presented here.

For each language, we run the neural network estimator multiple times with different random seeds, to control for variation in the random initialization of model parameters (see SI Section 3.2.3 for details).

In order to evaluate the average surprisal values S_t , we computed the empirical word-by-word surprisal values under the t th-order Markov model for held-out data, different from the data that was used to train the model. By evaluating on held-out data, we avoid underestimating the value of S_t due to overfitting. We chose held-out data based on existing splits of corpora, see section ‘Data’ below.

5.2 Data

We draw on syntactically annotated corpora, compiled by the Universal Dependencies project for several dozen languages (Nivre et al., 2017a). These are annotated in the format of Dependency Grammar (Hays, 1964; Hudson, 1984; Melčuk, 1988; Corbett et al., 1993; Tesnière and Kahane, 2015). In such dependency corpora, sentences are annotated with *dependency trees* (Figure 8). These are directed trees describing the grammatical relations among words. For example, the arcs labeled “obj” represent that the noun in question is the *direct object* of the verb, rather than e.g. the subject or an indirect object. A dependency arc is drawn from a *head* (e.g. the verb ‘has’) to a *dependent* (e.g. its object ‘book’). Dependency trees can be defined in terms of many different syntactic theories (Corbett et al., 1993). Although there are some differences in how different formalisms would draw trees for certain sentences, there is broad enough agreement about

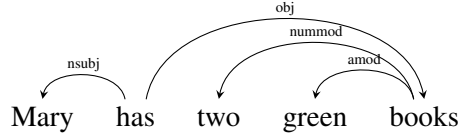


Figure 8: An English sentence with dependency annotations, according to the Universal Dependencies 2.4 standard (Nivre et al., 2017a). We visualize grammatical relations as arcs drawn from heads (e.g., the verb ‘has’) to dependents (e.g., its object ‘book’).

dependency trees that it has been possible to develop large-scale dependency-annotated corpora of text from dozens of languages (Nivre et al., 2017b).

We computed memory–surprisal tradeoffs for all languages for which there are Universal Dependencies 2.4 treebanks with a total of at least 500 sentences of training data. We excluded data from historical languages, as these corpora often include poetry, translated text, or texts spanning several centuries.² This resulted in 54 languages.

For each of these languages, we pooled all available corpora into one dataset. We excluded corpora that primarily contain code-switched text³ or text created by non-native speakers.⁴ Most Universal Dependencies corpora have a predefined split into *training*, *held-out* (also known as *development*), and *test* partitions. In most cases, we used the predefined data split, separately pooling data from the different partitions. For some languages with little data, there is no predefined training partition, or the training partition is smaller than the other partitions. In these cases, we redefined the split to obtain more training data. For these languages, we pooled all the available partitions, used 100 randomly selected sentences as held-out data, and used the remainder as training data.⁵ We did not make use of the *test* partitions here. We provide the sizes of the resulting datasets in SI Section 3.1. The datasets ranged in size from 564 sentences (Armenian) to 114,304 sentences (Czech), with a median of 5,255 sentences per language. For each language, we obtain a stationary process by concatenating the sentences from the corpus in random order, separated with an end-of-sentence symbol.

5.3 Defining baselines

Testing the Efficient Tradeoff Hypothesis requires comparing the memory–surprisal tradeoffs of real grammars to those of baseline grammars. The baseline grammars we construct are counterfactual ordering grammars that define consistent ordering rules similar to those found in actual languages (Figure 9). For instance, these grammars specify which dependents precede or follow their heads (e.g., whether objects follow or precede verbs, whether adjectives follow or precede nouns), and the relative order of different dependents on the same side of the head (e.g., whether noun phrases have order adjective-numeral-noun or numeral-adjective-noun). Our formalism of ordering grammars was introduced in Hahn et al. (2020), adapting the method of Gildea and Temperley (2007, 2010); Gildea and Jaeger (2015) to the setting of dependency corpora.

Universal Dependencies 2.4 defines 37 universal syntactic relations that are used to label dependency arcs across all corpora. These relations encode cross-linguistically meaningful relations such as subjects (*nsubj*, see Figure 8), objects (*obj*), and adjectival modifiers (*amod*). We define ordering grammars by as-

²Historical languages excluded: Ancient Greek, Classical Chinese, Coptic, Gothic, Latin, Old Church Slavonic, Old French.

³Hindi English corpus

⁴ESL for English, CFL for Chinese.

⁵This affects Amharic, Armenian, Breton, Buryat, Cantonese, Faroese, Kazakh, Kurmanji, Naija, Thai, and Uyghur.

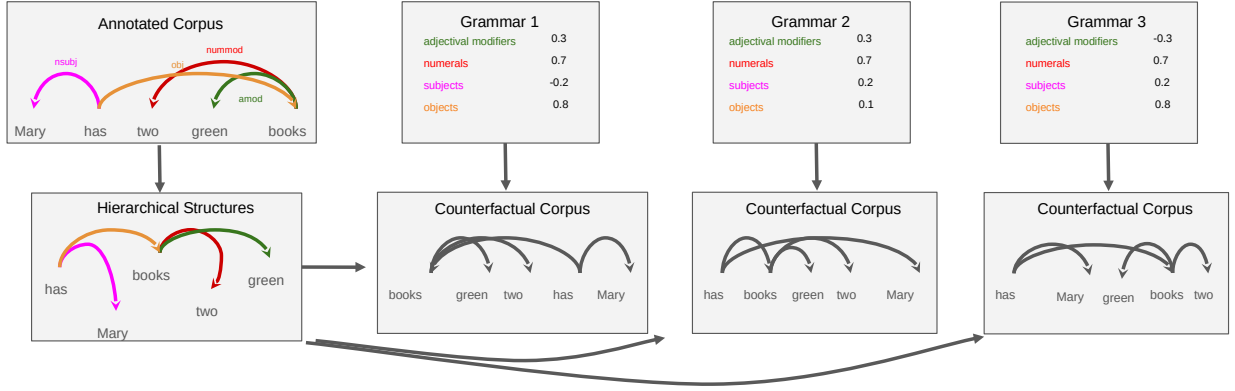


Figure 9: Estimating chance by constructing counterfactual grammars and languages: We start from an annotated dependency corpus of sentences annotated with syntactic dependencies (top left). We then extract the raw dependency structures, stripping away word order information (bottom left). We construct baseline ordering grammars that provide rules for ordering the words in such dependency structures (Grammars 1–3). Applying any such grammar to the dependency structures yields a counterfactual corpus of a hypothetical language that has the same dependency structures as the actual language, but different word order rules.

signing a parameter $a_\tau \in [-1, 1]$ to every one of these 37 universal syntactic relations. Relations sometimes have language-specific subtypes; we do not distinguish these subtypes. Following Gildea and colleagues, this parameter defines how dependents are ordered relative to their head: Given a head and a set of dependents, we order each dependent by the parameter a_τ assigned to the syntactic relation linking it to the head. Dependents with negative weights are placed to the left of the head; dependents with positive weights are placed to the right. Ordering grammars describe languages that have consistent word order. For instance, the subject is consistently ordered before or after the verb, depending on whether the parameter a_{nsubj} for the verb-subject dependency is positive or negative.

We constructed baseline grammars by randomly sampling the parameters a_τ . Such baseline grammars define languages that have consistent word order but, crucial for our purposes, do not exhibit systematic preferences for specific word order patterns such as short dependencies.

We first constructed at least 10 baseline grammars for each of the 54 real languages. We then continued to construct baseline grammars until a precision-based stopping criterion was reached. This criterion was designed to ensure that enough grammars were sampled to reliably compare the tradeoff curves of real and baseline grammars, without biasing results towards or against our hypothesis (see SI Section 3.2.3). The stopping criterion compared what fraction of baseline grammars had strictly more (or strictly less) efficient tradeoff curves than the real ordering, and required a bootstrapped 95% confidence interval for that ratio to have width ≤ 0.15 . The resulting number of baseline grammars ranged from 10 (Italian and Romanian) to 347 (Estonian).⁶

Due to the way ordering grammars are specified, certain kinds of rules cannot be modeled by our word order grammars. This includes rules sensitive to the category of the dependent, such as the difference between postverbal nominal objects and preverbal pronominal objects in Romance languages. It also includes

⁶Due to a scripting error, 846 grammars were generated for Erzya even though this was not required by the stopping criterion.

rules sensitive to larger context, e.g., the alternation between verb-final order in embedded clauses and verb-initial/verb-medial order in main clauses in German and Dutch. Furthermore, the model does not allow rules specifying interactions between different constituents, for instance, verb-second order, where exactly one dependent precedes the verb, and all others follow it. Finally, the model does not account for word order freedom, as all ordering choices are deterministic. In this sense, ordering grammars only represent an approximation to the kinds of ordering rules found in natural language (Gildea and Temperley, 2007, 2010; Gildea and Jaeger, 2015). Other models described in the literature (Futrell and Gibson, 2015; Wang and Eisner, 2016) mostly share these limitations.

To ensure that results are not due to the representational restrictions of the word order grammar formalism, we also compared the baselines to the result of ordering the corpora according to grammars that approximate the real orders to the extent possible in the grammar formalism. These grammars have exactly the same representational constraints as the baseline grammars while approximating the real orderings. We expect these grammars to have better memory–surprisal tradeoffs than comparable random baseline grammars across all languages. We created these ordering grammars by fitting them to the actual orderings of each language using the method of Hahn et al. (2020). They match the order of the actual language in those cases where order of a relation is fully consistent; for relations where order is variable, they approximate this by modeling the most frequent order. In representing word order rules, they have the same limitations as baseline grammars have, for instance, they cannot specify rules sensitive to the category of the dependent or to larger context.

5.4 Results

To test the Efficient Tradeoff Hypothesis, we compare the tradeoff curves for the real orders with those for random baseline grammars. In Figure 10, we show the estimated values of I_t for real and fitted orders and the median of I_t across different baseline grammars. In most languages, I_1 is distinctly larger for the actual and fitted orderings compared to the baseline orderings. This means that real orderings tend to concentrate more predictive information at the immediately preceding word than baseline grammars.

In Figure 11, we show the resulting bounds on the memory–surprisal tradeoff curves, showing surprisals at given levels of memory, for real and baseline languages. We compute surprisal at 40 evenly spaced points of memory (selected individually for each language, between 0 and the maximal memory capacity H_M obtained using Theorem 2), over real orders and baseline grammars. At each point, we then compute the median surprisal over all model runs for the real language, and over all baselines grammars. For each point, we compute an non-asymptotic and non-parametric 95% confidence interval for this median surprisal using the binomial test.

Numerically, the real language provides a better tradeoff than the median of the baselines across all languages, with four exceptions (Latvian, North Sami, Polish, Slovak). In order to quantify the degree of optimality of real orders, we further computed the area under the memory–surprisal tradeoff curve (AUC) for real and baseline orderings. Area under the curve (AUC) is a general quantity evaluating the efficiency of a tradeoff curve (Bradley, 1997). A *smaller* area indicates a *more efficient* memory–surprisal tradeoff. In Figure 12, we plot the AUC for the real orderings, together with the distribution of AUCs for baseline grammars. We quantify the degree of optimality by the fraction of baseline grammars for which the AUC is higher than for the real orders: The real ordering is highly efficient if it results in a lower AUC than almost all baseline grammars. Numerically, the AUC is smaller in the real orderings than in at least 50% of baseline grammars in all but three languages (Polish, Slovak, North Sami). We evaluated significance using a two-sided binomial test. In these three languages, the AUC is higher in the real orderings than in significantly less than 50% ($p < 0.01$ in each language). In all other languages except for Latvian, the fraction of more

efficient baseline grammars was significantly less than 50%, at $p = 0.01$, where we applied Hochberg’s step-up procedure (Hochberg, 1988) to control for multiple comparisons. In 42 of the 54 languages, the real language was more efficient than all of the sampled baseline grammars.

The AUC for the fitted grammars is lower than more than 50% of random baseline grammars in all 54 languages ($p < 0.01$, using two-sided Binomial test and Hochberg’s step-up procedure). Thus, we replicate the result that ordering regularities of real languages provide more efficient tradeoffs than most possible order grammars even when comparing within the same word order grammar formalism.

5.5 Discussion

We have found that 50 out of 54 languages provide better memory–surprisal tradeoffs than random baselines with consistent but counterfactual word order rules. Four languages provide exceptions; these are Latvian (Baltic), North Sami (Uralic), Polish and Slovak (both Slavic). These four languages did not have significantly lower AUC values than half of the random baselines. One feature that unites these four languages is that they have strong word order freedom, as we will see below in Figure 13. Word order freedom plausibly makes sentences less predictable, as the same syntactic structure can receive different surface realizations. We thus hypothesized that word order freedom impacts the memory–surprisal tradeoff, and that languages with more strongly fixed word order should display more optimal memory–surprisal tradeoffs.

To test this hypothesis, we examined the correlation between word order freedom and the surprisal difference between real and baseline orderings. To quantify word order freedom, we used a corpus-based estimate, the **branching direction entropy** (Futrell et al., 2015b). This is the entropy of the ordering (head-first or dependent-first) of dependencies conditioned on the dependency label and the part-of-speech label of head and dependent. These two quantities are plotted in Figure 13. We found that branching direction entropy was strongly correlated with the surprisal difference between real and baseline orderings (Spearman correlation -0.58 , $p < .0001$).

This result might mean that optimization of word orders for memory–surprisal tradeoffs is indeed stronger in languages with more fixed word order, and that word order freedom leads to less efficient memory–surprisal tradeoffs. A second possibility is that languages with seemingly free word order encode other information in word order, in particular, information about information structure (e.g. Givon, 1988; Firbas, 1966, 1974; Myhill, 1985). Next, we test the latter hypothesis by examining whether the degree of optimization changes when taking into account information structure.

5.6 Controlling for Information Structure

In this section we address the question about word order freedom raised in the last section. We draw on a corpus of Czech with information structure annotation to determine whether real orders are more optimized when comparing to baselines taking information structure into account.

Languages with flexible word order often show a strong influence of information structure on word order (Givon, 1988; Jacobs, 1988; Neeleman and van de Koot, 2016). Due to the difficulty of annotating information structure, only relatively few datasets have annotations for information structure, and even fewer datasets have both syntactic and information structure annotation. We draw on the Prague Dependency Treebank of Czech (Böhmová et al., 2003; Mikulová et al., 2006), which has both types of annotation. Czech is a language with relatively high degree of word order freedom, which is generally thought to be strongly impacted by information structure (Firbas, 1966, 1974).

About one third of the Prague Dependency Treebank has annotation for topic-focus articulation (Mikulová et al., 2006). Constituents are annotated for contrastiveness and for contextual boundedness, i.e., givenness.

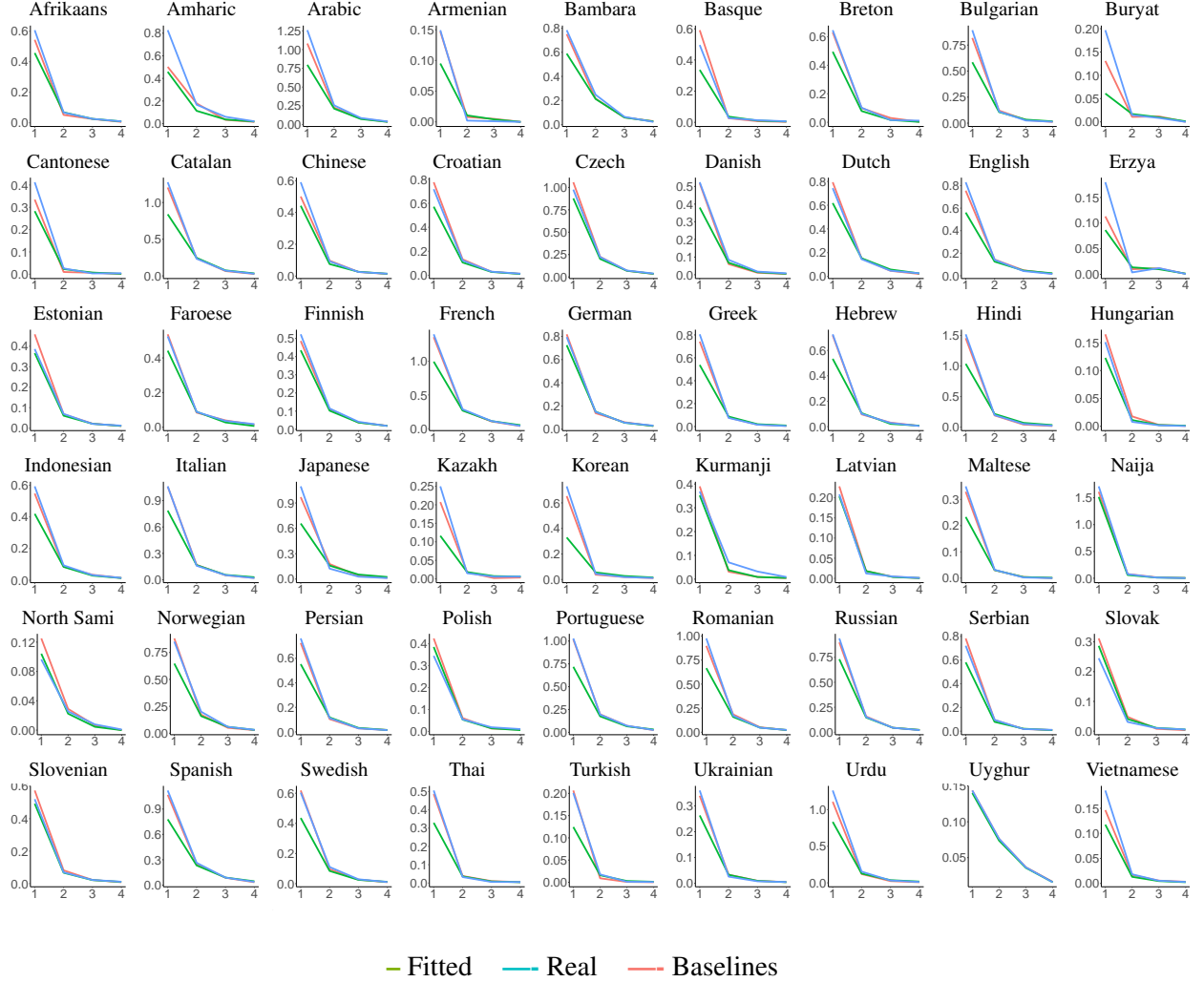


Figure 10: Conditional mutual information I_t (y-axis) as a function of t (x-axis), for real, fitted and baseline orders. We plot the median over all sampled baseline grammars.

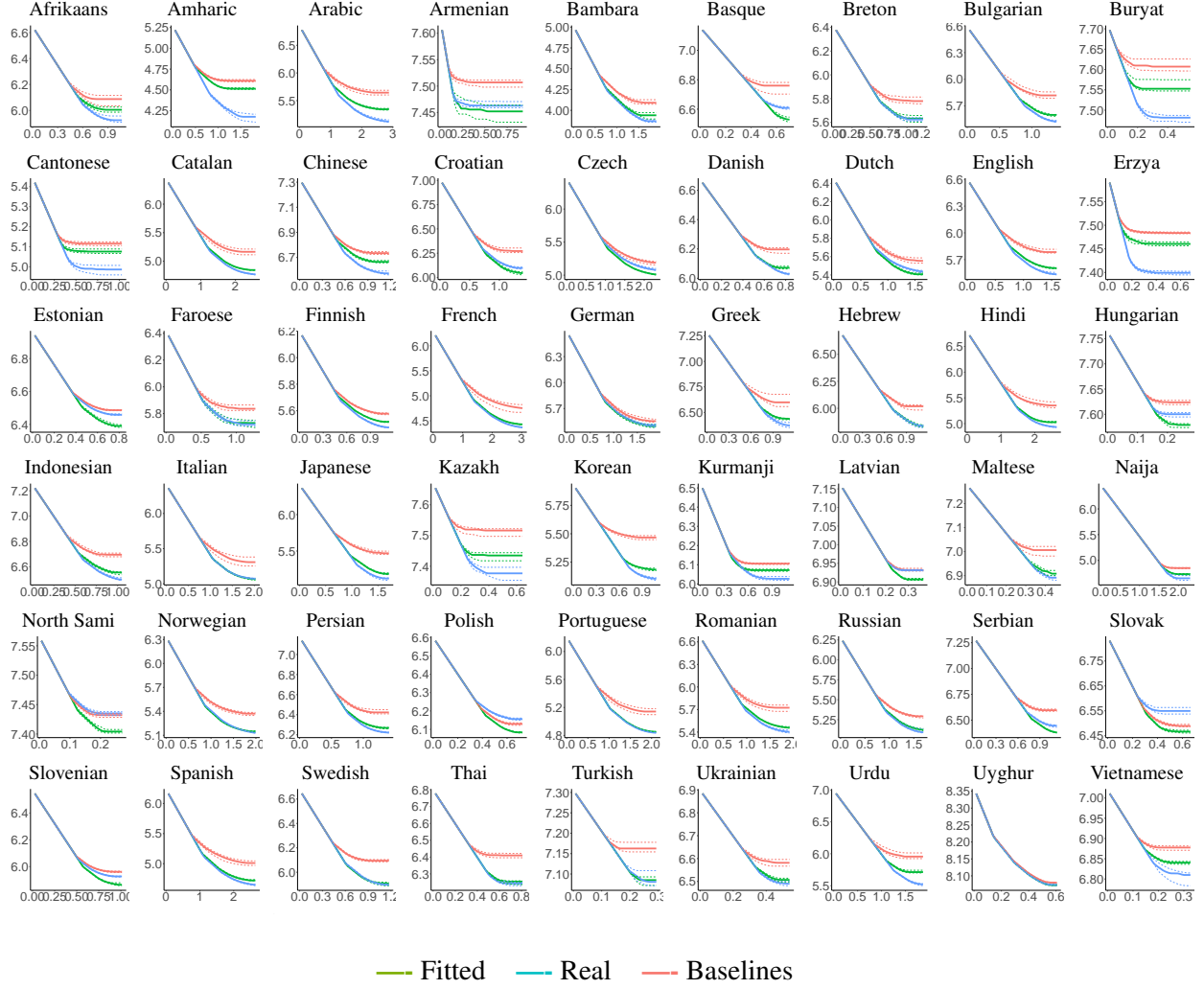


Figure 11: Surprisal (y-axis) at given memory level (x-axis), for real, fitted, and baseline orders. For the real and fitted orders, we provide the median across multiple random seeds of the neural network estimator for I_t (see SI Section 3.2.2), and 95% confidence bands. For the baseline grammars, we provide the median across both the sampled baseline grammars and multiple random seeds of the estimator, and 95% confidence bands for this median.

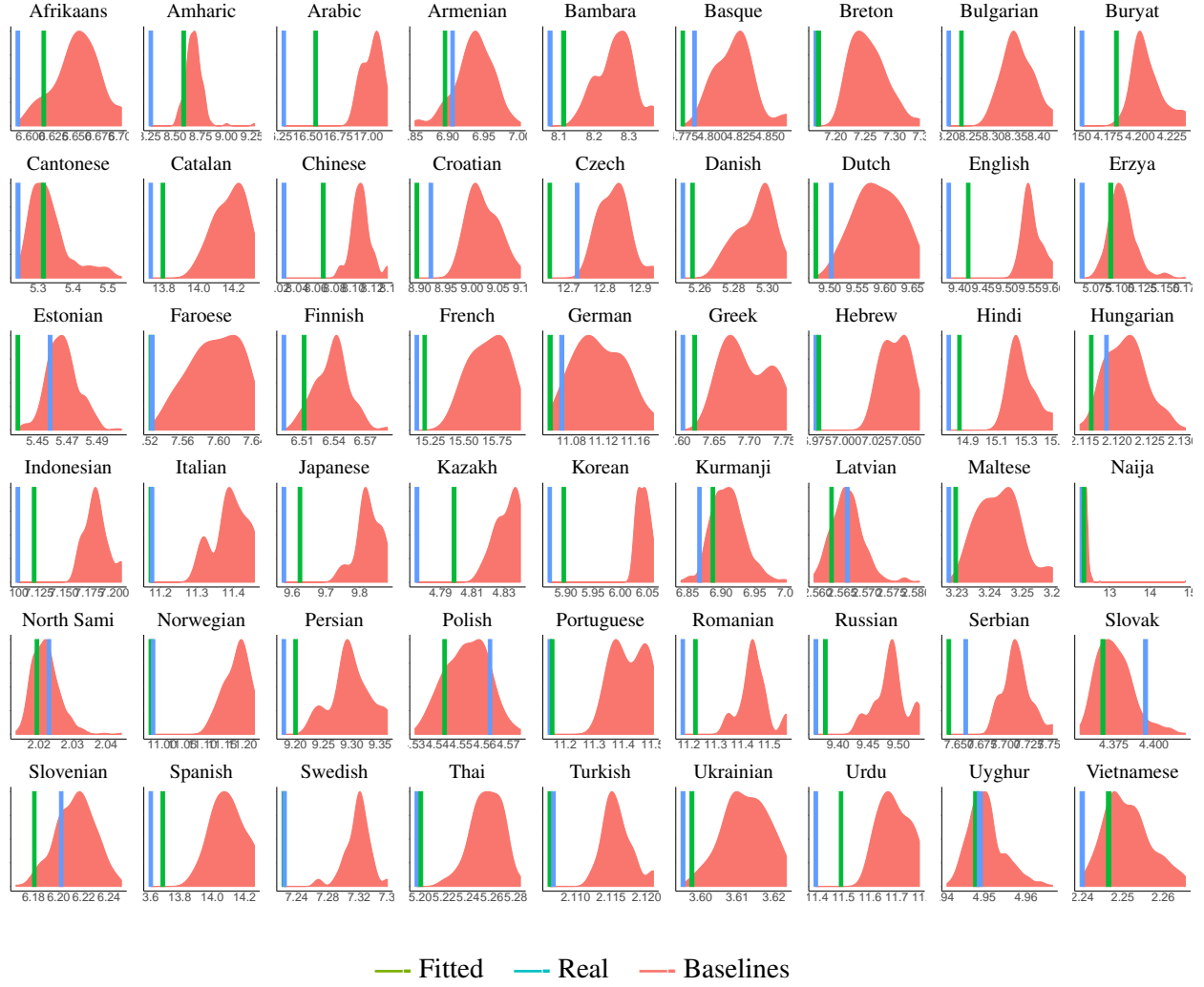


Figure 12: Histograms for the Area under the Curve (AUC) for the memory–surprisal tradeoffs for real, fitted, and random orders. We provide a kernel density smoothing estimate of the distribution of random baseline orders. A smaller AUC value indicates a more efficient tradeoff. In most cases, the real and fitted orders provide more efficient tradeoffs than most or all baseline grammars.

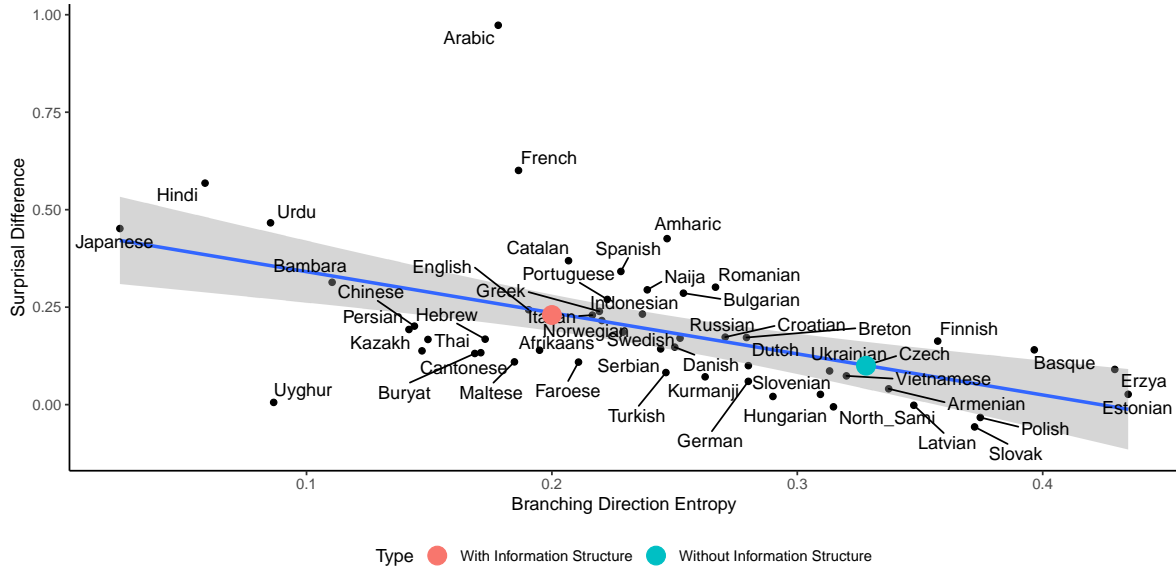


Figure 13: Word order freedom and strength of optimization: For each of the 54 languages, we show word order freedom as measured by branching entropy, and the difference between the real order’s surprisal and the median surprisal of the baseline grammars, at the maximum memory value (see Figure 11). Languages with higher branching direction entropy show a smaller reduction in surprisal compared to baseline orders. For Czech, we also provide an estimate accounting for information structure (red dot), see below, ‘Controlling for Information Structure’, for more information.

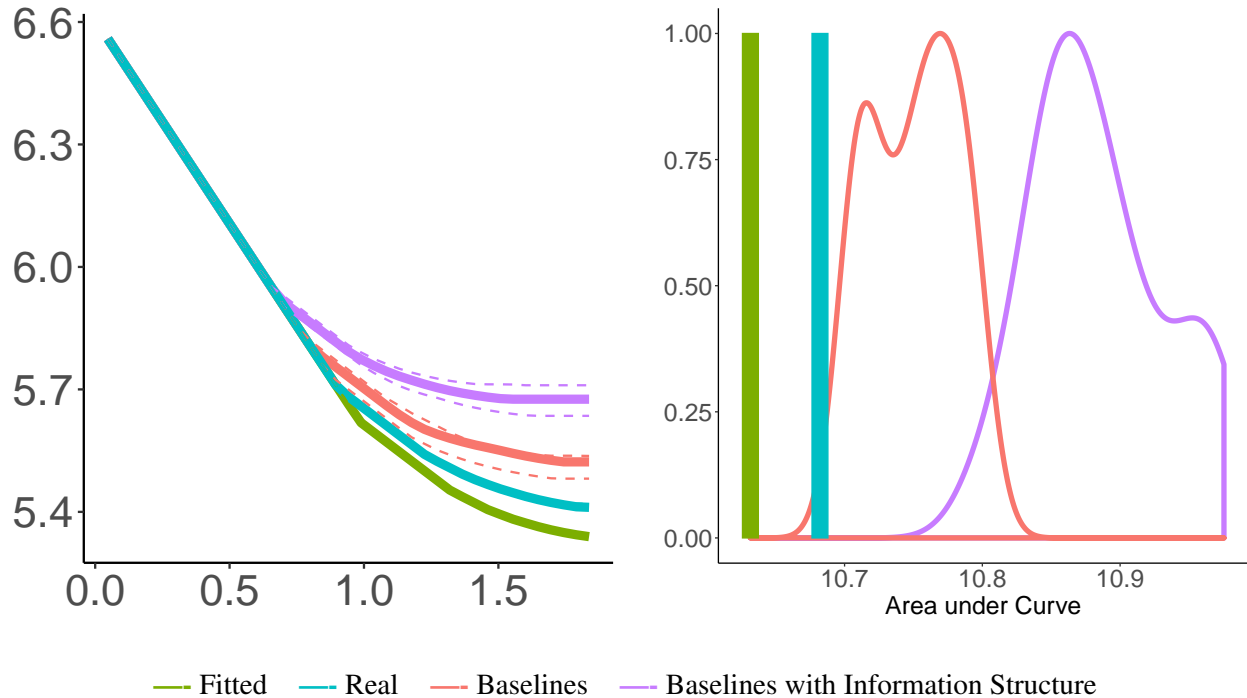


Figure 14: Left: Memory-surprisal tradeoff for Czech with information structure. Right: AUC for Czech, for baselines without information structure and baselines with information structure. Optimization of real orders is stronger when considering information structure in baselines.

Contextually bound expressions are presumed as given in context so that their referent is uniquely determined by the context; contextually bound expressions are contrastive if they choose from a contextually given set of alternatives (Mikulová et al., 2006, Section 10.2). Three labels are used: “c” for contrastive and contextually bound, “f” for contextually non-bound, “t” for non-contrastive contextually bound. These labels were diagnosed based on constituent order and intonation. Some constituents remain unmarked, the vast majority of which are function words such as adpositions, conjunctions, and auxiliaries; we introduce a label “NA” for these. To define baselines, we extend the word order grammar formalism by defining separate weights for each combination of the 37 syntactic relations and these four information structure labels.

We obtained 38,727 training sentences and 5,228 held-out sentences. We created 20 baseline grammars with information structure, 20 baseline grammars without it.

Results We show estimated tradeoffs and the distributions over AUC values in Figure 14. As this experiment was conducted only on the subset of the Prague Dependency Treebank that has information structure annotation, the numerical values are slightly different from those in Figure 11. We compare the real orders both with the same baselines as above, and with the baselines taking information structure into account. Baselines show a larger gap in efficiency between real and baseline grammars when the baselines condition word order on information structure. This suggests that, among word orders that encode information structure, the real order of Czech provides a very efficient memory–surprisal tradeoff, and that the strength of optimization is underestimated when comparing against baselines that do not take information structure into account.

In Figure 13, we show how the data point for Czech changes when including information structure in the word order modeling. When modeling information structure, branching direction entropy decreases, while the surprisal difference between real and baseline orders increases. This suggests that the weaker optimization in free word order languages observed in Study 2 might in part be because ordering grammars do not take information structure into account. In general, we expect that conditioning word order on more sources of information will increase the set of possible word orders, and thus decrease predictability and increase surprisal. As more corpora become available, it will be important to reproduce this finding on data from further languages. If this finding replicates, then this would mean that the impact of order freedom on the strength of optimization observed in Study 2 is an artifact of the fact that languages differ in the degree to which their word order encodes information structure, and that similar degrees of optimization might actually hold across such different languages.

Discussion Using data from Czech, we found that the difference between the memory–surprisal tradeoffs of real and baseline orders increases if we choose baseline orderings that encode information structure, as real orders do. We hypothesize that this in part explains why the observed strength of memory efficiency optimization is negatively correlated with the degree of word order freedom: Languages with flexible word order typically encode information structure in word order, which increases average surprisal. This does not mean that conditioning word order on information structure makes language less efficient in general. Rather, encoding information structure in word order may increase the information content transmitted to the listener, which may in turn balance an increase in surprisal processing effort (Hahn et al., 2020). Due to the difficulty and cost of annotating information structure, we could only evaluate this hypothesis on data from one language. As more annotated data becomes available, this should be replicated on data from further languages.

5.7 Interim summary

In this section, we tested the Efficient Tradeoff Hypothesis on dependency corpora from 54 languages, comparing observed word orders to hypothetical baseline grammars. We found that, in 50 out of 54 languages, real orders provide more efficient memory–surprisal tradeoffs than most baseline grammars. This result also held when comparing real and baseline orderings within a single grammar formalism. These results suggest that, across languages, word order favors information locality more strongly than most possible alternative orders.

We also found that the degree of optimization was weaker in languages with high degrees of word order freedom. Using data from Czech that is annotated for both syntax and information structure, we provided evidence that this dependence on word order freedom is an artifact of the fact that languages with flexible word order tend to encode information structure in word order.

Taken together, Studies 1 and 2 suggest that crosslinguistic word orders are in part impacted by a pressure towards efficient memory–surprisal tradeoffs, and thus information locality. To test whether the Efficient Tradeoff Hypothesis holds at different levels of representation, we consider morpheme order in Study 3.

6 Study 3: Morpheme Order

The Efficient Tradeoff Hypothesis should apply not just at the level of words, but at the level of any linguistic element. For instance, just as observed word orders exhibit information locality, the order of morphemes within words should also be structured so that morphemes which predict each other are close to each other.

Here, we apply the Efficient Tradeoff Hypothesis to predict the order of morphemes within morphologically complex words in two agglutinative languages. We study two agglutinative languages for which extensive corpora with hand-annotated morphological segmentation and labeling are available: Japanese and Sesotho. We compare the memory–surprisal tradeoff of the actual morpheme orders in these languages with hypothetical baseline orderings. Furthermore, we construct hypothetical orderings that are optimized for the efficiency of the memory–surprisal tradeoff, and compare these to the actual morpheme orderings, to investigate whether morpheme order in these languages can be predicted by optimization of tradeoff efficiency. Below, we first give brief sketches of the morphological patterns in these languages.

Verb Suffixes in Japanese In Japanese, verbs are marked with an extensive number of suffixes. For example, the following verb forms are marked with multiple suffixes:

- (1) a. mi rare mash yoo
 see PASSIVE POLITENESS FUTURE
 ‘will be seen’
 b. mi taku nakat ta
 see DESIDERATIVE NEGATION PAST
 ‘did not wish to see’

Based on corpus data and the linguistic literature on Japanese, we identified the following frequent verb suffixes, occurring in the following order outwards from the verb root (see SI for details).

1. *suru*: obligatory suffix after Sino-Japanese words when they are used as verbs
2. Valence: causative (-*ase*-) (Hasegawa (2014, 142), Kaiser et al. (2013, Chapter 13))
3. Voice and Mood: passive (-*are*-, -*rare*-) (Hasegawa (2014, 152), Kaiser et al. (2013, Chapter 12)) and potential (-*e*-, -*are*-, -*rare*-) (Kaiser et al., 2013, 398)
4. Politeness (-*mas*-) (Kaiser et al., 2013, 190).
5. Mood: desiderative (-*ta*-) (Kaiser et al., 2013, 238)
6. Negation (-*n*-)
7. Tense, Aspect, Mood, and Finiteness: past (-*ta*), future/hortative (-*yoo*) (Kaiser et al., 2013, 229), nonfiniteness (-*te*) (Kaiser et al., 2013, 186)

Verb Affixes in Sesotho Sesotho (also known as Southern Sotho) is a Southern Bantu language spoken primarily in Lesotho and South Africa. Sesotho verbs are marked with both prefixes and suffixes (Demuth, 1992). Common prefixes include markers for agreement with subjects and objects; object prefixes always follow subject prefixes (2-a). Common suffixes include markers changing valence and voice, and a mood suffix (2-b).

- (2) a. oa di rek a
 SUBJECT.AGREEMENT OBJECT.AGREEMENT buy INDICATIVE
 ‘(he) is buying (it)’ (Demuth, 1992)

- b. o pheh el a
 SUBJECT.AGREEMENT cook APPLICATIVE INDICATIVE
 ‘(he) cooks (food) for (him)’ (Demuth, 1992)

We identified affix morphemes and their ordering based on the analysis in Demuth (1992), supplemented with information from grammars of Sesotho (Doke and Mofokeng, 1967; Guma, 1971). See SI for details. We identified the following prefixes:

1. Subject agreement: This morpheme encodes agreement with the subject, for person, number, and noun class (the latter only in the 3rd person) (Doke and Mofokeng, 1967, §395). The annotation provided by Demuth (1992) distinguishes between ordinary subject agreement prefixes and agreement prefixes used in relative clauses; we distinguish these morpheme types here.
2. Negation (Doke and Mofokeng, 1967, §429)
3. Tense/aspect marker (Doke and Mofokeng, 1967, §400–424)
4. Object agreement or reflexive marker (Doke and Mofokeng, 1967, §459). Similar to subject agreement, object agreement denotes person, number, and noun class features of the object.

We identified the following suffixes:

1. Semantic derivation: reversive (e.g., ‘do’ → ‘undo’) (Doke and Mofokeng, 1967, §345)
2. Valence: Common valence-altering suffixes include causative, neuter/stative, applicative, and reciprocal (Doke and Mofokeng, 1967, §307–338). See SI for details on their meanings.
3. Voice: passive (Doke and Mofokeng, 1967, §300)
4. Tense (Doke and Mofokeng, 1967, §369)
5. Mood (Doke and Mofokeng, 1967, §386–445)
6. Interrogative and relative markers, appended to verbs in certain interrogative and relative clauses (Doke and Mofokeng, 1967, §160, 271, 320, 714, 793).

6.1 Experiment

Data Selection and Processing For Japanese, we drew on Universal Dependencies data. In the tokenization scheme used for Japanese, most affixes are separated as individual tokens, effectively providing morpheme segmentations. We used the GSD corpus, Version 2.4, (Tanaka et al., 2016; Asahara et al., 2018), as it was the only corpus with a training set and freely available word forms. In the corpus, verb suffixes largely correspond to auxiliaries (with tag `AUX`); only a few morphemes tagged `AUX` are not standardly treated as suffixes (see SI), and one frequent suffix (*-te*) is labeled `CONJ`. We selected verb forms by selecting all chains of a verb (tag `VERB`) followed by any number of auxiliaries (tag `AUX`) from the training set of the corpus. When the suffix *-te* (tag `CONJ`) followed such a chain, we added this. We labeled suffixes for underlying morphemes with the help of the lemmatization provided for each suffix in the corpus (see SI Section 4.3 for details). The passive and potential (slot 3) markers are formally indistinguishable for many verbs. As we cannot systematically distinguish them on the basis of the available corpus annotation, we merge these into a single underlying morpheme ‘Passive/Potential’.

We obtained 15,281 verb forms in the training set and 1,048 verb forms in the held-out set. Of the forms in the training set, 27% had two or more suffixes (modal group: two suffixes, accounting for 20% of forms; maximum seven suffixes). While predicting order naturally focuses on datapoints with more than one suffix, we include the other datapoints for estimating conditional mutual information I_t .

For Sesotho, we used the Demuth Corpus (Demuth, 1992) of child and child-directed speech, containing about 13K utterances with 500K morphemes. The corpus has very extensive manual morphological segmentation and annotation; each verb form is segmented into morphemes, which are annotated for their function. Sesotho verbs carry both prefixes and suffixes. We extracted 37K verb forms (see SI 4.2 for details). We randomly selected 5% to serve as held-out data and used the remaining 95% as training data. 93% of forms had two or more affixes (modal group: three affixes, accounting for 36% of forms; maximum eight affixes).

Estimating Memory-Surprisal Tradeoff We modeled incremental prediction on the level of morpheme sequences. To do so, we represented each verb form as a sequence of a stem and affix morphemes, abstracting away from morphophonemic interactions between neighboring morphemes. As in many languages, affixes in Japanese and Sesotho show morphophonemic interactions between neighboring morphemes; for instance, the Japanese politeness morpheme *-mas-* takes the form *-masu* when it is word-final, while it has the allomorph *-mase-* when followed by the negation suffix *-n*. Modeling prediction on the level of morphemes, as opposed to phonemes, controls for these interactions.⁷

In analogy to Studies 1–2, we modeled verb forms as a stationary stochastic process by concatenating the verb forms from the corpus in random order.

We calculated I_t by estimating an n -gram model on the training set and then computing the average surprisal S_t as cross-entropy on the held-out set using Kneser-Ney smoothing. The model may overfit as the context size t increases, leading to higher cross-entropies for larger values of t . We mitigated overfitting for large t by estimating

$$\hat{S}_t := \min_{s \leq t} S_s, \quad (13)$$

where S_s is the cross-entropy of the s 'th order Markov model on held-out data. This procedure ensures that \hat{S}_t can only decrease as the context size t increases.

Parameterizing Alternative Orderings We parameterized alternative affix orderings by assigning a weight in $[0, 1]$ to each morpheme. Given such a grammar, affixes are ordered by the values assigned to their underlying morphemes. We considered all morphemes annotated in the corpora, including low-frequency ones going beyond the ones identified above (see SI for details).

To verify that this formalism is appropriate for capturing morpheme order in Japanese and Sesotho, we fitted models parameterized in this way to the observed orders. Ordering morphemes according to these fitted models recovered the observed order for almost all forms (98.6 % for Japanese, 99.93% for Sesotho prefixes, 97.4% for Sesotho suffixes). Exceptions largely concern low-frequency affixes beyond those considered here. We take this as confirmation that the formalism is generally suited to capture morpheme order.

For each language, we constructed 40 baseline grammars by randomly sampling weights.

Creating Optimal Orders In order to create optimal orders to compare real orders to, we optimized orderings for the AUC under the memory–surprisal tradeoff curve with an adaptation of the hill climbing method that Gildea and Jaeger (2015) used to optimize word order grammars for the length of syntactic dependencies and trigram surprisal.

⁷See SI Section 4.3 for qualitatively similar results when modeling prediction at the phoneme level.

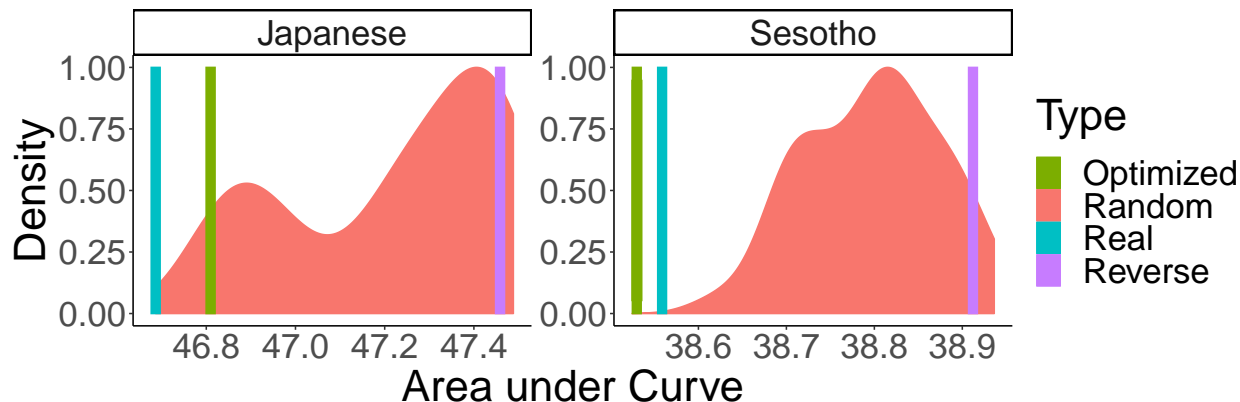


Figure 15: Areas under the curve for the memory–surprisal tradeoff for verb affixes in Japanese (left) and Sesotho (right). For the baseline grammars, we show a Kernel Density estimate. In both Japanese and Sesotho, real and optimized orderings lead to lower AUC than all of the 40 baseline samples.

We randomly initialized the assignment of weights to morphemes, and then iteratively change the assignment to reduce AUC. In each iteration, we randomly chose one morpheme, and evaluate AUC for each way of ordering it between two other morphemes. We then updated the weights to the ordering that yields the lowest AUC. To speed up optimization, we restricted to morphemes occurring at least 10 times in the corpus for 95% of iterations, and to 10% of possible orderings in each step. These choices vastly reduced computation time by reducing time spent on low-frequency morphemes. This optimization method is approximate, as it only guarantees convergence to a local optimum (Gildea and Jaeger, 2015), not to a globally optimal assignment.

We ran this method for 1,000 iterations. Empirically, AUC converged after a few hundred iterations. To control for the randomness in initialization and the optimization steps, we ran this algorithm ten times. Different runs achieved almost the same AUC values (SD 0.0051 in Japanese, 0.0036 in Sesotho). For Sesotho, we ran the algorithm separately for prefixes and suffixes due to computational efficiency considerations.⁸

6.2 Results

In Figure 15, we compare the area under curve of the memory–surprisal tradeoff for Japanese and Sesotho verb forms under different orderings. Both observed orders and the approximately optimized grammars show lower AUCs than all 40 random baseline samples, in both languages. For comparison, we also show AUC for the order resulting from *reversing* all suffix chains in the observed orders; this results in high AUC even exceeding most random grammars. These results show that Japanese and Sesotho affix orderings enable approximately optimal memory–surprisal tradeoffs.

We now ask to what extent the observed morpheme ordering is predicted correctly by approximately optimized grammars. In Table 1, we give summary statistics about the accuracy of optimized grammars in predicting affix order in the corpus, together with random baseline figures. We evaluate accuracy using two methods: In one method (‘Pairs’), we consider, for each verb form in the corpus, all pairs of prefixes (or suffixes). We report the proportion of these pairs in the corpus for which the relative order of the two affixes

⁸With the exception of the tense/aspect markers, none of the morpheme types discussed above can occur both as prefixes and suffixes. Therefore, we do not expect this separation to impact results.

		Prefixes		Suffixes	
		Pairs	Full	Pairs	Full
Japanese	Optimized	–	–	0.953 (SD 0.011)	0.943 (SD 0.014)
	Baseline	–	–	0.497 (SD 0.287)	0.425 (SD 0.29)
Sesotho	Optimized	0.988 (SD 0.0)	0.989 (SD 0.0)	0.756 (SD 0.014)	0.676 (SD 0.017)
	Baseline	0.672 (SD 0.305)	0.604 (SD 0.338)	0.423 (SD 0.204)	0.332 (SD 0.211)

Table 1: Accuracy of approximately optimized orderings, and of random baseline orderings, in predicting verb affix order in Japanese and Sesotho. ‘Pairs’ denotes the rate of pairs of morphemes that are ordered correctly, and ‘Full’ denotes the rate of verb forms where order is predicted entirely correctly. We show means and standard deviations over ten different runs of the optimization algorithm (‘Optimized’), and over different random orderings (‘Random’).

	Real	Optimized
	Stem	Stem
1	suru	suru
2	causative	causative
3	passive/potential	passive/potential
4	desiderative	negation
5	politeness	future
6	negation	politeness
7	future	desiderative
	past	nonfinite
	nonfinite	past

Table 2: Comparing order of Japanese affixes in the observed orders (left) and according to an approximately optimized grammar (right). We organize the affixes in the real order into the seven slots described above.

is as predicted by the grammar. In the other method, (‘Full’), we report the proportion of verb forms in the corpus that has exactly the affix ordering predicted by the grammar. In both measures, we average over all ten approximately optimized grammars for each language.

Japanese results. In Japanese, by both measures, optimized grammars recover the observed orders with high accuracy. We compare the real grammar with the approximately optimized grammar that achieved the lowest AUC value in Table 2. We conducted an error analysis comparing the real Japanese morpheme order against our approximately optimized orders. We extracted the pairs of morphemes whose relative order is incorrectly predicted, excluding pairs involving low-frequency morphemes not discussed here. Results are shown in Table 3. The most frequent divergence for this grammar is that politeness and negation suffixes are consistently ordered incorrectly; this affects 74 corpus examples (out of 15K total examples).

We also found that prediction was more accurate when modeling on the level of phonemes, suggesting that divergence between model predictions and actual order might be related to phonological pressure (see SI Section 4.3).

Sesotho results. We compare the real Sesotho grammar with the approximately optimized grammar that achieved the lowest AUC value in Table 4. In Sesotho, for prefixes, all optimized grammars almost exactly

Error		Frequency
politeness	negation	74
desiderative	negation	14
politeness	future	9

Table 3: Errors in Japanese: We show pairs of morphemes that are ordered incorrectly by the approximately optimized grammar with the lowest AUC value. We indicate the number of such pairs occurring in the corpus. We only show errors where both morphemes are among the high-frequency ones studied here.

	Real	Optimized
1	Subject	Subject
	Subject (rel.)	Subject (rel.)
2	Negation	Tense/aspect
3	Tense/aspect	Negation
4	Object	Object
	Stem	Stem
1	Reversive	Passive
2	Causative	Reciprocal
	Neuter	Tense/aspect
	Applicative	Neuter
	Reciprocal	Relative
3	Passive	Causative
4	Tense/aspect	Applicative
5	Mood	Interrogative
6	Interrogative	Reversive
	Relative	Mood

Table 4: Comparing order of Sesotho affixes in the observed orders (left) and according to an approximately optimized grammar (right). Note that order was separately optimized for prefixes and suffixes.

recover the ordering described above. The only divergence among the high-frequency morphemes is that negation and the tense/aspect prefix are ordered incorrectly; this accounts for only 12 occurrences in the data set, as the two prefixes rarely co-occur (Table 5, top).

For Sesotho suffixes, order is recovered at above-chance accuracies (Table 1, bottom), though with some divergences. The most common error (Table 5, bottom) is that relative and interrogative suffixes are consistently placed closer to the verb stem than the mood suffix. We conjecture that this happens because all Sesotho verbs uniformly have a mood suffix, suggesting that there might be lower mutual information between the stem and the mood suffix than between the stem and these two suffixes. Furthermore, valence-changing suffixes are ordered farther away from the stem than various other suffixes, in contrast with the actual orders. Interestingly, we found that prediction was more accurate in this respect when estimating I_t naively on the training set (see SI Section 4.3), suggesting that the available corpus data does not sufficiently determine the optimal ordering.

Error		Frequency
Negation	Tense/aspect	12

Error		Frequency
Mood	Interrogative	2204
Mood	Relative	858
Applicative	Tense/aspect	347
Causative	Tense/aspect	174
Neuter	Tense/aspect	155
Reversive	Causative	100
Applicative	Passive	81
Causative	Passive	61
Applicative	Relative	49
Causative	Relative	41

Table 5: Errors in Sesotho prefixes (top) and suffixes (bottom). We show the ten most common errors where both morphemes are among the high-frequency ones studied here.

6.3 Discussion

We have found that the ordering of verb affixes in Japanese and Sesotho provides approximately optimal memory–surprisal tradeoffs, close to the efficiency of orderings computationally optimized for efficiency. We further found that parts of these languages’ ordering rules can be derived from optimizing order for efficient tradeoffs.

Here we argue that the memory–surprisal tradeoff provides an explanation of previously-existing typological generalizations, and an operationalization of previous functionally-motivated explanations for them; in particular, we argue that the notion of mutual information operationalizes the concept of ‘relevance.’

One prominent typological generalization due to Bybee (1985) claims that there exists a universal ordering of verbal inflectional morphemes across languages:

verb stem valence voice aspect tense mood modality subj. person subj. number

Morphemes are claimed either to go in the order above (suffixes), or its reverse (prefixes). This hierarchy makes no statements as to which affixes are realized as prefixes or suffixes.

Japanese and Sesotho verb affixes are broadly in agreement with Bybee’s generalization. For instance, valence and voice suffixes are closer to the stem than tense/aspect/mood markers. Subject agreement in Sesotho is farther away from the verb than tense/aspect/mood prefixes. This ordering is reproduced closely by optimization in Japanese and for Sesotho prefixes, and to some extent also for Sesotho suffixes.

Bybee (1985, p. 37) argues further that morpheme order is determined by the degree of *relevance* between the affix and the stem, that is, the degree to which “the semantic content of the first [element] directly affects or modifies the semantic content of the second” (p. 13). She argues that elements whose meanings are more relevant to each other appear closer together. For instance, the meaning of a verb is impacted more strongly by a causative affix than by a tense affix: Combining a verb with a causative marker results in a form that denotes a different action, whereas a tense affix only locates the action in time.

We conjecture that this notion of relevance is related to mutual information. If an affix has a stronger impact on the meaning of the verb, it will typically not be applicable to all verbs. For instance, causative

markers will only attach to verbs whose semantics is compatible with causation. In contrast, a past tense marker can attach to all verbs that are compatible with actions that can have occurred in the past. Therefore, we expect that affixes that are more relevant to a verb stem will also tend to have higher mutual information with the verb stem. If they have higher mutual information with the verb stem, then the principle of information locality predicts that they will go close to the verb stem.

7 General Discussion

We introduced a notion of memory efficiency in language processing: the memory–surprisal tradeoff. We then tested the resulting Efficient Tradeoff Hypothesis: Order of elements in natural language is characterized by efficient memory–surprisal tradeoffs, compared to other possible orders. In Study 1, we showed that the Efficient Tradeoff Hypothesis predicts the known preference for short dependencies. In Study 2, we used corpus data from 54 languages to show that real word orders provide more efficient tradeoffs than baseline order grammars. In Study 3, we showed that in two languages (Japanese and Sesotho) the order of verb affixes not only provides approximately optimal tradeoffs, but can also partly be predicted by optimizing for the efficiency of the memory–surprisal tradeoff.

Here, we discuss the limitations of our results and the implications they have more broadly for the fields of psycholinguistics, typology, and information theory.

7.1 Role of Comprehension, Production, and Acquisition

Our results leave open the causal mechanism leading to the observed optimization, in particular, whether optimization is the result of minimizing effort during comprehension, production, or acquisition. One possibility is that optimization reflects an effort on the side of the speaker to produce utterances that are easy to comprehend by listeners, a strategy known as *audience design* (Clark and Murphy, 1982; Lindblom, 1990; Brennan and Williams, 1995). More efficient memory–surprisal tradeoffs are useful from the listener’s perspective because they allow for better prediction with lower memory investment than less efficient tradeoffs.

Another possibility is that optimization reflects production-internal pressures to minimize effort on the speaker’s part during sentence planning (Bock and Warren, 1985; Ferreira and Dell, 2000; MacDonald, 2013; Fedzechkina and Jaeger, 2020). That is, instead of speakers optimizing for the benefit of listeners, the iterated application of production-internal heuristics that reduce speaker effort may result in more efficient tradeoffs (MacDonald, 2013). While our theory is stated in terms of the efficiency of language processing for a comprehender of language, we can show that an analogous memory–surprisal tradeoff exists in language production, and that speakers with bounded memory capacity can minimize production errors when the language has stronger information locality (see SI Section 1.4 for a mathematical statement and proof).

Finally, optimization may reflect biases that come into play during language learning. It is possible that memory efficiency makes languages more learnable, as learning should require less memory resources for languages with more efficient memory–surprisal tradeoffs. Evidence from artificial language learning experiments suggests that language acquisition is biased towards efficiency in communication and processing (e.g. Fedzechkina et al., 2012, 2017).

7.2 Relation to Models of Sentence Processing

There is a substantial literature proposing sentence processing models and quantitative memory metrics for sentence processing. In this section, we discuss how our theoretical results relate to and generalize these

previously proposed models. We do not view our model as competing with or replacing any of these models; instead, our information-theoretic analysis captures aspects that are common to most of these models and shows how they arise from very general modeling assumptions.

In the Information Locality Bound Theorem, we proved a formal relationship between the entropy of memory H_M and average surprisal S_M . We made no assumptions about the architecture of incremental memory, and so our result is general across all such architectures. Memory representations do not have to be rational or optimal for our bound in Theorem 2 to hold. There is no physically realizable memory architecture that can violate this bound.

However, psycholinguistic theories may differ on whether the entropy of memory H_M really is the right measure of memory load, and on whether average surprisal S_M really is the right predictor of processing difficulty for humans. Therefore, in order to establish that our information-theoretic processing model generalizes previous theories, we will establish two links:

- Our measure of memory usage generalizes theories that are based on counting numbers of objects stored in incremental memory (e.g., Yngve, 1960; Miller and Chomsky, 1963; Frazier, 1985; Gibson, 1998; Koble et al., 2013; Graf and Marcinek, 2014; Graf et al., 2015; Gerth, 2015; Graf et al., 2017; De Santo, 2020). Furthermore, for theories where memory is constrained in its capacity for *retrieval* rather than storage (e.g., McElree et al., 2003; Lewis and Vasishth, 2005), the information locality bound will still hold.
- Our predictor of processing difficulty (i.e., average surprisal) reflects at least a *component* of the predicted processing difficulty under other theories.

Below, we discuss the connections between our theory and existing theories of human sentence processing with regard to the points above.

Storage-Based Theories There is a long tradition of models of human language processing in which difficulty is attributed to high working memory load. These models go back to Yngve (1960)’s production model, where difficulty was associated with moments when a large number of items have to be kept on a parser stack; this model correctly predicted the difficulty of center-embedded clauses, but problematically predicted that left-branching structures should be hard (Kimball, 1973). Other early examples include Miller and Chomsky (1963) and Frazier (1985)’s measure of syntactic complexity based on counting the number of local nonterminal nodes. More recently, a line of literature has formulated complexity metrics based on how many nodes are kept in incremental memory for how long during parsing, and used linear or ranked combinations of these metrics to predict acceptability differences in complex embeddings (Koble et al., 2013; Graf and Marcinek, 2014; Rambow and Joshi, 2015; Graf et al., 2015; Gerth, 2015; Graf et al., 2017; De Santo, 2020).

Our measure of memory complexity—i.e., the memory entropy H_M —straightforwardly generalizes measures based on counting items stored in memory. If each item stored in memory requires k bits of storage, then storing n items would require a capacity of nk bits in terms of memory entropy H_M . In general, if memory entropy is H_M and all items stored in memory take k bits each to store, then we can store H_M/k items. However, the memory entropy H_M is more general as a measure of storage cost, because it allows that different items stored in memory might take different numbers of bits to store, and also that the memory representation might be able to compress the representations of multiple items when they are stored together, so that the capacity required to store two items might be less than the sum of the capacity required to store each individual item. Previous work has argued that visual working memory is characterized by an information-

theoretic capacity limit (Brady et al., 2008; Sims et al., 2012); we extend this idea to incremental memory as used in language processing.

The Dependency Locality Theory The connection with the Dependency Locality Theory is particularly interesting. Our lower bound on memory usage, described in Theorem 2 Eq. 11, is formally similar to Storage Cost in the Dependency Locality Theory (DLT) (Gibson, 1998, 2000). In that theory, storage cost at a given timestep is defined as the *number of predictions* that are held in memory. Our bound on memory usage is stated in terms of mutual information, which indicates the *amount of predictive information* extracted from the previous context and stored in memory. As the notion of ‘number of predictions’ is subsumed by the notion of ‘amount of predictive information’, our measure generalizes DLT storage cost.

The other component of the DLT is integration cost, the amount of difficulty incurred by establishing a long-term syntactic dependency. In our framework, DLT integration cost corresponds to surprisal given an imperfect memory representation, following Futrell et al. (2020).

There is one remaining missing link between our theory of processing difficulty and theories such as the Dependency Locality Theory: our information locality theorem says that *statistical* dependencies should be short-term, whereas psycholinguistic theories of locality have typically focused on the time-span of **syntactic dependencies**: words which depend on each other to determine the meaning or the well-formedness of a sentence. Statistical dependencies, in contrast, mean that whenever one element of a sequence determines or predicts another element *in any way*, those two elements should be close to each other in time.

If statistical dependencies, as measured using mutual information, can be identified with syntactic dependencies, then that would mean that information locality is straightforwardly a generalization of dependency locality. Futrell et al. (2019) give theoretical and empirical arguments that this is so. They show that syntactic dependencies as annotated in dependency treebanks identify word pairs with especially high mutual information, and give a derivation showing that this is to be expected according to a formalization of the postulates of dependency grammar. The connection between mutual information and syntactic dependency has also been explored in the grammar induction literature (de Paiva Alves, 1996; Yuret, 1998).

Cue-Based Retrieval Models In some psycholinguistic theories, memory-related difficulty arises not because of a bound on memory capacity, but rather because of difficulties involved in retrieving information from memory (McElree, 2000; Lewis and Vasishth, 2005; Nicenboim and Vasishth, 2018; Vasishth et al., 2019). We are able to prove an analogous theorem that applies to such theories: see SI Section 1.3 for an extension of our analysis to the case involving a short-term memory (STM) with unbounded capacity, a working memory (WM) with limited capacity, and cost associated with communication between WM and STM. Essentially, the constraint on the memory state in our theorem above can be re-interpreted as a constraint on the capacity of a communication channel linking STM to WM. In particular, this result constrains average surprisal for memory models based on cue-based retrieval such as the ACT-R model of Lewis and Vasishth (2005).

The ACT-R model of Lewis and Vasishth (2005) does not have an explicit surprisal cost. Instead, surprisal effects are interpreted as arising because, in less constraining contexts, the parser is more likely to make decisions that then turn out to be incorrect, leading to additional correcting steps. We view this as an algorithmic-level implementation of a surprisal cost: If a word w_t is unexpected given the current state of the working memory, then its current state must provide insufficient information to constrain the actual syntactic state of the sentence, meaning that the parsing steps made to integrate w_t are likely to include more backtracking and correction steps. Thus, we argue that cue-based retrieval models predict that the surprisal $-\log P(w_t|m_t)$ will be part of the cost of processing word w_t .

Mutual information, as we defined it in Eq. 10, is the *average* pointwise mutual information over an entire probability distribution.

Our information locality bound theorem differs from this previous result in three ways:

1. Futrell et al. (2020) required an assumption that incremental memory is subject to decay over time. In contrast, we do not require any assumptions about incremental memory except that it has bounded capacity (or that retrieval operations have bounded capacity; see above).
2. Our result is a precise bound, whereas the previous result was an approximation based on neglecting higher-order interactions among words.
3. Our result is about the fall-off of the mutual information between words, *conditional on the intervening words*. The previous result was about the fall-off of *pointwise* mutual information between specific words, without conditioning on the intervening words.

As noted above, this previous work defined information locality in terms of the *unconditional* (pointwise) mutual information between linguistic elements. Here we argue that conditional mutual information is more relevant for measuring memory usage than unconditional mutual information. While the decay of conditional mutual information provably provides a lower bound on memory entropy, the decay of unconditional mutual information does not. In SI Section 2.3, we provide an example of a stochastic process where unconditional mutual information does not decay with distance, but memory requirements remain low.

Experience-Based and Connectionist Models Our model and results are compatible with work arguing that memory strategies adapt to language structure and language statistics, and that experience shapes memory performance in syntactic processing (e.g. MacDonald and Christiansen, 2002; Wells et al., 2009). For instance, MacDonald and Christiansen (2002) argue for a connectionist model in which network structure and language experience account for processing capacity. Such models use recurrent neural networks with some fixed number of neurons, which can be understood as a specific kind of constrained memory (Futrell et al., 2020). A case in point is the observation that forgetting effects in nested head-final dependencies are reduced or absent in head-final structures (Vasishth et al., 2010; Frank et al., 2016; Frank and Ernst, 2019), which has been modeled using connectionist models (Engelmann and Vasishth, 2009; Frank et al., 2015b), which can be interpreted as modeling surprisal conditioned on imperfect memory (Futrell et al., 2020).

7.3 Limitations

Finiteness of Data As corpora are finite, estimates for I_t may not be reliable for larger values of t . In particular, we expect that models will underestimate I_t for large t , as models will not be able to extract and utilize all available information over longer distances. This means that we might not be able to consistently estimate the asymptotic values of the average surprisal S_M as the memory capacity goes to infinity, i.e. the entropy rate S_∞ .

This has implications for the interpretation of the memory–surprisal tradeoffs at higher values of memory entropy H_M . In Study 2, the lowest achieved surprisals are different for real and baseline orderings. This does not necessarily mean that these orderings really have different entropy rates S_∞ . It is logically possible that real and baseline languages actually have the same entropy rate S_∞ , but that baseline orderings spread the same amount of predictive information over a larger distance, making it harder for models to extract given finite corpus data. What our results do imply is that real languages provide lower surprisals in the setting of relatively small memory budgets. This result only depends on the estimates of I_t for small values of t , which are most trustworthy.

Nature of the Bound Our theoretical result provides a lower bound on the tradeoff curve that holds across all ways of physically realizing a memory representation obeying the postulates (1–3). However, this bound may be loose in two ways: First, architectural properties of human memory might introduce additional constraints on possible representations. Depending on the role played by factors other than information-theoretic capacity, the tradeoffs achieved by these human memory representations need not be close to achieving the theoretical bounds. Furthermore, depending on properties of the stochastic process, the bound might be loose across all models, that is, there are processes where the bound is not attainable by any memory architecture. This can happen if there is strong uncertainty as to which aspects of the past observations will be relevant to the future. We provide an artificial example with analytical calculations in SI Section 2.1; however, this example does not seem linguistically natural.

Extralinguistic Context Comprehension Postulate 1 states that the memory state after receiving a word is determined by that word and the memory state before receiving this word. The assumption about information flow disregards the role of information sources that are external to the linguistic material in the sentence. For instance, the interlocutors might have common knowledge of the weather, and the listener might use this to construct predictions for the speaker’s utterances, even if no relevant information has been mentioned in the prior discourse. Such sources of information are disregarded in our model. They are also disregarded in many other models of memory in sentence processing. Taking extralinguistic context into account would likely result in more efficient tradeoffs, as this can introduce additional cues helping to predict the future better.

Limitations of Baseline Language Grammar Model In Study 2, baseline grammars are constructed in a formalism that cannot fully express some word order regularities found in languages. For instance, it cannot express orders that differ in main clauses and embedded clauses (see discussion there for further limitations). These limitations are common to most other order grammar formalisms considered in the literature; despite these limitations, such word order models have demonstrated reasonably good fits to corpus data on word order and human judgments of fluency (Futrell and Gibson, 2015; Wang and Eisner, 2016). These limitations do not affect the estimated tradeoffs of real orders. However, the grammar model determines the baseline distribution, and thus impacts the comparison with real orders. For example, to the extent that strict word order decreases surprisal, this baseline distribution will put more weight on relatively efficient baselines, potentially resulting in a smaller difference with real orders than for baseline distributions that allow more flexibility. We emphasize that this limitation does not hold in Study 3, where the formalism provides very close fit to observed morpheme orders.

7.4 Information-theoretic studies of language

Our work opens up a connection between psycholinguistics, linguistic typology, and statistical studies of language. Here, we survey the connections between our work and previous statistical studies.

Our formalization of memory is related to studies of dynamic systems in the physics literature. Our memory–surprisal curve is closely related to the *predictive information bottleneck* introduced by Still (2014) and studied by Marzen and Crutchfield (2016); in particular, it is a version of the *recursive information bottleneck* (Still, 2014, §4). In the limit of optimal prediction and minimal surprisal, our formalization of memory cost is equivalent to the notion of *Statistical Complexity* (Crutchfield and Young, 1989; Shalizi and Crutchfield, 2001); this is the minimum value of H_M that achieves $S_M = S_\infty$. In the limit $T \rightarrow \infty$, the quantity in Eq. 11 is equal to the *excess entropy* (Crutchfield and Young, 1989). The link between memory

and information locality provided by our Theorem 2 appears to be a novel theoretical contribution. Related to Theorem 2, Sharan et al. (2016) show a link between excess entropy and approximability by n -th order Markov models, noting that processes with low excess entropy can be approximated well with Markov models of low order.

Debowski (2011) studied the excess entropy of language across very long ranges of text, in particular aiming to determine whether it is finite. Our work contrasts with this work in that we are interested in dependencies within sentences. Hahn and Futrell (2019) empirically estimate the predictive rate-distortion tradeoff of natural language (Still, 2014; Marzen and Crutchfield, 2016). This tradeoff is related to but different from the memory–surprisal tradeoff: it models a setting where past observations are compressed once as a whole, while a memory capacity bottleneck holds at every point in time in our model. Nonetheless, their method might also be applicable to estimating memory–surprisal tradeoff curves.

The average surprisal of real and counterfactual word orders have been studied by Gildea and Jaeger (2015) and Hahn et al. (2020). Gildea and Jaeger (2015) found that, in five languages, real orders provide lower trigram surprisal than baseline languages. This work can be viewed as instantiating our model in the case where the encoding function M records exactly the past two words, and showing that these five languages show optimization for surprisal under this encoding function. Hahn et al. (2020) compared surprisal and parseability for real and baseline orders as estimated using neural network models, arguing that word orders optimize a tradeoff between these quantities. The results of Experiment 2 complement this by showing that real word orders optimize surprisal across possible memory capacities and memory encoding functions.

8 Conclusion

In this work, we have provided evidence that human languages order elements in a way that reduces cognitive resource requirements, in particular memory effort. We provided an information-theoretic formalization of memory requirements as a tradeoff of memory and surprisal. We showed theoretically that languages have more efficient tradeoffs when they show stronger degrees of information locality. Information locality provides a formalization of various locality principles from the linguistic literature, including dependency locality (Gibson, 1998), domain minimization (Hawkins, 2004), and the proximity principle (Givón, 1985). Using this result, we provided evidence that languages order words and morphemes in such a way as to provide efficient memory–surprisal tradeoffs.

Our result shows that wide-ranging principles of order in natural language can be explained from highly generic cognitively-motivated information-theoretic principles. The locality properties we have discussed are some of the most characteristic properties of natural language, setting natural language apart from other codes studied in information theory. Therefore, our result raises the question of whether other distinctive characteristics of language—for example, mildly context-sensitive syntax, duality of patterning, and compositionality—might also be explained in terms of information-theoretic resource constraints on production and comprehension.

References

- Abney, S. P. and Johnson, M. (1991). Memory requirements and local ambiguities of parsing strategies. *Journal of Psycholinguistic Research*, 20(3):233–250.
- Altmann, G. T. and Kamide, Y. (1999). Incremental interpretation at verbs: Restricting the domain of subsequent reference. *Cognition*, 73(3):247–264.

- Arnold, J. E., Losongco, A., Wasow, T., and Ginstrom, R. (2000). Heaviness vs. newness: The effects of structural complexity and discourse status on constituent ordering. *Language*, 76(1):28–55.
- Asahara, M., Kanayama, H., Tanaka, T., Miyao, Y., Uematsu, S., Mori, S., Matsumoto, Y., Omura, M., and Murawaki, Y. (2018). Universal dependencies version 2 for japanese. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*.
- Balling, L. W. and Kizach, J. (2017). Effects of surprisal and locality on danish sentence processing: An eye-tracking investigation. *Journal of psycholinguistic research*, 46(5):1119–1136.
- Bartek, B., Lewis, R. L., Vasishth, S., and Smith, M. R. (2011). In search of on-line locality effects in sentence comprehension. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 37(5):1178.
- Behaghel, O. (1932). Deutsche Syntax, vol. 4. *Heidelberg: Winter*.
- Bentz, C., Alikaniotis, D., Cysouw, M., and Ferrer-i Cancho, R. (2017a). The entropy of words—learnability and expressivity across more than 1000 languages. *Entropy*, 19(6):275.
- Bentz, C., Alikaniotis, D., Samardžić, T., and Buttery, P. (2017b). Variation in word frequency distributions: Definitions, measures and implications for a corpus-based language typology. *Journal of Quantitative Linguistics*, 24(2-3):128–162.
- Berger, T. (2003). Rate-distortion theory. *Wiley Encyclopedia of Telecommunications*.
- Berwick, R. C. and Weinberg, A. (1984). *The grammatical basis of linguistic performance*. Cambridge, MA: MIT Press.
- Bock, J. K. and Warren, R. K. (1985). Conceptual accessibility and syntactic structure in sentence formulation. *Cognition*, 21(1):47–67.
- Böhmová, A., Hajič, J., Hajičová, E., and Hladká, B. (2003). The prague dependency treebank. In *Treebanks*, pages 103–127. Springer.
- Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern recognition*, 30(7):1145–1159.
- Brady, T. F., Konkle, T., and Alvarez, G. A. (2008). Efficient coding in visual short-term memory: Evidence for an information-limited capacity. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 30.
- Brady, T. F., Konkle, T., and Alvarez, G. A. (2009). Compression in visual working memory: Using statistical regularities to form more efficient memory representations. *Journal of Experimental Psychology: General*, 138(4):487.
- Brennan, S. E. and Williams, M. (1995). The feeling of another’s knowing: Prosody and filled pauses as cues to listeners about the metacognitive states of speakers. *Journal of memory and language*, 34(3):383–398.
- Bybee, J. L. (1985). *Morphology : a study of the relation between meaning and form*. John Benjamins, Amsterdam.

- Chomsky, N. (2005). Three factors in language design. *Linguistic Inquiry*, 36(1):1–61.
- Clark, H. H. and Murphy, G. L. (1982). Audience design in meaning and reference. In *Advances in psychology*, volume 9, pages 287–299. Elsevier.
- Corbett, G. G., Fraser, N. M., and McGlashan, S. (1993). *Heads in Grammatical Theory*. Cambridge University Press, Cambridge.
- Cover, T. M. and Thomas, J. (2006). *Elements of Information Theory*. John Wiley & Sons, Hoboken, NJ.
- Crutchfield, J. P. and Young, K. (1989). Inferring statistical complexity. *Physical Review Letters*, 63(2):105–108.
- de Paiva Alves, E. (1996). The selection of the most probable dependency structure in Japanese using mutual information. In *34th Annual Meeting of the Association for Computational Linguistics*, pages 372–374.
- De Santo, A. (2020). MG parsing as a model of gradient acceptability in syntactic islands. In *Proceedings of the Society for Computation in Linguistics*, volume 3.
- Debowski, L. (2011). Excess entropy in natural language: Present state and perspectives. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(3):037105.
- Demberg, V. and Keller, F. (2009). A computational model of prediction in human parsing: Unifying locality and surprisal effects. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 31.
- Demuth, K. (1992). Acquisition of Sesotho. In *The cross-linguistic study of language acquisition*, pages 557–638. Lawrence Erlbaum Associates.
- Doke, C. M. and Mofokeng, S. M. (1967). *Textbook of southern Sotho grammar*. Longmans.
- Doob, J. L. (1953). *Stochastic processes*. New York Wiley.
- Engelmann, F. and Vasishth, S. (2009). Processing grammatical and ungrammatical center embeddings in English and German: A computational model. In *Proceedings of the Ninth International Conference on Cognitive Modeling, Manchester, UK*, pages 240–45.
- Fedorenko, E., Woodbury, R., and Gibson, E. (2013). Direct evidence of memory retrieval as a source of difficulty in non-local dependencies in language. *Cognitive Science*, 37(2):378–394.
- Fedzechkina, M., Chu, B., and Jaeger, T. F. (2017). Human information processing shapes language change.
- Fedzechkina, M. and Jaeger, T. F. (2020). Production efficiency can cause grammatical change: Learners deviate from the input to better balance efficiency against robust message transmission. *Cognition*, 196:104115.
- Fedzechkina, M., Jaeger, T. F., and Newport, E. L. (2012). Language learners restructure their input to facilitate efficient communication. *Proceedings of the National Academy of Sciences*, 109(44):17897–17902.
- Ferreira, V. S. and Dell, G. S. (2000). Effect of ambiguity and lexical availability on syntactic and lexical production. *Cognitive psychology*, 40(4):296–340.

- Firbas, J. (1966). On defining the theme in functional sentence analysis. *Travaux Linguistiques de Prague*, 1:267–280.
- Firbas, J. (1974). Some aspects of the czechoslovak approach to problems of functional sentence perspective. In Danes, F., editor, *Papers on Functional Sentence Perspective*, pages 11–37.
- Frank, S. L. and Bod, R. (2011). Insensitivity of the Human Sentence-Processing System to Hierarchical Structure. *Psychological Science*, 22(6):829–834.
- Frank, S. L. and Ernst, P. (2019). Judgements about double-embedded relative clauses differ between languages. *Psychological research*, 83(7):1581–1593.
- Frank, S. L., Otten, L. J., Galli, G., and Vigliocco, G. (2015a). The erp response to the amount of information conveyed by words in sentences. *Brain and language*, 140:1–11.
- Frank, S. L., Trompenaars, T., and Vasishth, S. (2015b). Cross-Linguistic Differences in Processing Double-Embedded Relative Clauses: Working-Memory Constraints or Language Statistics? *Cognitive Science*.
- Frank, S. L., Trompenaars, T., and Vasishth, S. (2016). Cross-linguistic differences in processing double-embedded relative clauses: Working-memory constraints or language statistics? *Cognitive Science*, 40(3):554–578.
- Frazier, L. (1985). Syntactic complexity. *Natural language parsing: Psychological, computational, and theoretical perspectives*, pages 129–189.
- Friston, K. (2010). The free-energy principle: a unified brain theory? *Nature reviews neuroscience*, 11(2):127–138.
- Futrell, R. (2019). Information-theoretic locality properties of natural language. In *Proceedings of the First Workshop on Quantitative Syntax (Quasy, SyntaxFest 2019)*, pages 2–15.
- Futrell, R. and Gibson, E. (2015). Experiments with generative models for dependency tree linearization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1978–1983, Lisbon, Portugal. Association for Computational Linguistics.
- Futrell, R., Gibson, E., and Levy, R. P. (2020). Lossy-context surprisal: An information-theoretic model of memory effects in sentence processing. *Cognitive science*, 44(3).
- Futrell, R., Mahowald, K., and Gibson, E. (2015a). Large-scale evidence of dependency length minimization in 37 languages. *Proceedings of the National Academy of Sciences*, 112(33):10336–10341.
- Futrell, R., Mahowald, K., and Gibson, E. (2015b). Quantifying word order freedom in dependency corpora. In *Proceedings of the Third International Conference on Dependency Linguistics (Depling 2015)*, pages 91–100.
- Futrell, R., Qian, P., Gibson, E., Fedorenko, E., and Blank, I. (2019). Syntactic dependencies correspond to word pairs with high mutual information. In *Proceedings of the fifth international conference on dependency linguistics (depling, syntaxfest 2019)*, pages 3–13.
- Gershman, S. J. (2020). Origin of perseveration in the trade-off between reward and complexity. *bioRxiv*.

- Gerth, S. (2015). *Memory Limitations in Sentence Comprehension: A Structural-based Complexity Metric of Processing Difficulty*, volume 6. Universitätsverlag Potsdam.
- Gibson, E. (1991). *A computational theory of human linguistic processing: Memory limitations and processing breakdown*. PhD thesis, Carnegie Mellon University Pittsburgh, PA.
- Gibson, E. (1998). Linguistic complexity: Locality of syntactic dependencies. *Cognition*, 68(1):1–76.
- Gibson, E. (2000). The dependency locality theory: A distance-based theory of linguistic complexity. In Marantz, A., Miyashita, Y., and O’Neil, W., editors, *Image, Language, Brain: Papers from the First Mind Articulation Project Symposium*, pages 95–126.
- Gibson, E., Futrell, R., Piandadosi, S. T., Dautriche, I., Mahowald, K., Bergen, L., and Levy, R. (2019). How efficiency shapes human language. *Trends in cognitive sciences*.
- Gibson, E. and Thomas, J. (1999). Memory limitations and structural forgetting: The perception of complex ungrammatical sentences as grammatical. *Language and Cognitive Processes*, 14(3):225–248.
- Gildea, D. and Jaeger, T. F. (2015). Human languages order information efficiently. *arXiv:1510.02823 [cs]*. arXiv: 1510.02823.
- Gildea, D. and Temperley, D. (2007). Optimizing Grammars for Minimum Dependency Length. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 184–191, Prague, Czech Republic. Association for Computational Linguistics.
- Gildea, D. and Temperley, D. (2010). Do Grammars Minimize Dependency Length? *Cognitive Science*, 34(2):286–310.
- Givón, T. (1985). Iconicity, isomorphism and non-arbitrary coding in syntax. *Iconicity in syntax*, pages 187–219.
- Givón, T. (1988). The pragmatics of word-order: Predictability, importance and attention. In Hammond, M., Moravcsik, E., and Wirth, J., editors, *Studies in Syntactic Typology*, pages 243–284.
- Givón, T. (1991). Markedness in grammar: distributional, communicative and cognitive correlates of syntactic structure. *Stud Lang*, 15:335–370.
- Goodkind, A. and Bicknell, K. (2018). Predictive power of word surprisal for reading times is a linear function of language model quality. In *Proceedings of the 8th Workshop on Cognitive Modeling and Computational Linguistics (CMCL 2018)*, pages 10–18.
- Graf, T., Fodor, B., Monette, J., Rachiele, G., Warren, A., and Zhang, C. (2015). A refined notion of memory usage for minimalist parsing. In *Proceedings of the 14th Meeting on the Mathematics of Language (MOL 2015)*, pages 1–14, Chicago, USA. Association for Computational Linguistics.
- Graf, T. and Marcinek, B. (2014). Evaluating evaluation metrics for minimalist parsing. In *Proceedings of the Fifth Workshop on Cognitive Modeling and Computational Linguistics*, pages 28–36.
- Graf, T., Monette, J., and Zhang, C. (2017). Relative clauses as a benchmark for Minimalist parsing. *Journal of Language Modelling*, 5:57–106.

- Greenberg, J. H. (1963). Some universals of grammar with particular reference to the order of meaningful elements. *Universals of language*, 2:73–113.
- Grodner, D. and Gibson, E. (2005). Consequences of the serial nature of linguistic input for sentential complexity. *Cognitive science*, 29(2):261–290.
- Guma, S. M. (1971). *An outline structure of Southern Sotho*. Shuter and Shooter.
- Gwilliams, L., King, J.-R., Marantz, A., and Poeppel, D. (2020). Neural dynamics of phoneme sequencing in real speech jointly encode order and invariant content. *bioRxiv*.
- Hahn, M. and Futrell, R. (2019). Estimating predictive rate-distortion curves via neural variational inference. *Entropy*, 21(7):640.
- Hahn, M., Jurafsky, D., and Futrell, R. (2020). Universals of word order reflect optimization for communicative efficiency. *PNAS*.
- Hale, J. T. (2001). A probabilistic Earley parser as a psycholinguistic model. In *Proceedings of the Second Meeting of the North American Chapter of the Association for Computational Linguistics and Language Technologies*, pages 1–8.
- Hasegawa, Y. (2014). *Japanese: A linguistic introduction*. Cambridge University Press.
- Hawkins, J. A. (1994). *A performance theory of order and constituency*. Cambridge University Press, Cambridge.
- Hawkins, J. A. (2003). Efficiency and complexity in grammars: Three general principles. *The nature of explanation in linguistic theory*, pages 121–152.
- Hawkins, J. A. (2004). *Efficiency and complexity in grammars*. Oxford University Press, Oxford.
- Hawkins, J. A. (2014). *Cross-linguistic variation and efficiency*. Oxford University Press, Oxford.
- Hays, D. G. (1964). Dependency theory: A formalism and some observations. *Language*, 40:511–525.
- Hochberg, Y. (1988). A sharper bonferroni procedure for multiple tests of significance. *Biometrika*, 75(4):800–802.
- Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.
- Hudson, R. A. (1984). *Word grammar*. Blackwell Oxford.
- Jacobs, J. (1988). Probleme der freien wortstellung im deutschen. *Sprache und Pragmatik. Arbeitsberichte*, 5:8–37.
- Jaeger, T. F. and Tily, H. J. (2011). On language ‘utility’: Processing complexity and communicative efficiency. *Wiley Interdisciplinary Reviews: Cognitive Science*, 2(3):323–335.
- Kaiser, S., Ichikawa, Y., Kobayashi, N., and Yamamoto, H. (2013). *Japanese: A comprehensive grammar*. Routledge.
- Kimball, J. (1973). Seven principles of surface structure parsing in natural language. *Cognition*, 2(1):15–47.

- Kobele, G. M., Gerth, S., and Hale, J. (2013). Memory resource allocation in top-down minimalist parsing. In *Formal Grammar*, pages 32–51. Springer.
- Kuperberg, G. R. and Jaeger, T. F. (2016). What do we mean by prediction in language comprehension? *Language, cognition and neuroscience*, 31(1):32–59.
- Landauer, R. (1961). Irreversibility and heat generation in the computing process. *IBM journal of research and development*, 5(3):183–191.
- Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*, 106(3):1126–1177.
- Levy, R. (2013). Memory and surprisal in human sentence comprehension. In van Gompel, R. P. G., editor, *Sentence Processing*, page 78–114. Hove: Psychology Press.
- Lewis, R. L. and Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29(3):375–419.
- Lin, H. W. and Tegmark, M. (2017). Critical Behavior in Physics and Probabilistic Formal Languages. *Entropy*, 19(7):299.
- Lindblom, B. (1990). On the communication process: Speaker-listener interaction and the development of speech. *Augmentative and Alternative Communication*, 6(4):220–230.
- Liu, H. (2008). Dependency distance as a metric of language comprehension difficulty. *Journal of Cognitive Science*, 9(2):159–191.
- Liu, H., Xu, C., and Liang, J. (2017). Dependency distance: A new perspective on syntactic patterns in natural languages. *Physics of Life Reviews*.
- MacDonald, M. C. (2013). How language production shapes language form and comprehension. *Frontiers in psychology*, 4:226.
- MacDonald, M. C. and Christiansen, M. H. (2002). Reassessing working memory: Comment on just and carpenter (1992) and waters and caplan (1996).
- Marzen, S. E. and Crutchfield, J. P. (2016). Predictive Rate-Distortion for Infinite-Order Markov Processes. *Journal of Statistical Physics*, 163(6):1312–1338.
- McElree, B. (2000). Sentence comprehension is mediated by content-addressable memory structures. *Journal of psycholinguistic research*, 29(2):111–123.
- McElree, B., Foraker, S., and Dyer, L. (2003). Memory structures that subserve sentence comprehension. *Journal of Memory and Language*, 48(1):67–91.
- Melčuk, I. A. (1988). *Dependency syntax: Theory and practice*. SUNY Press.
- Mikulová, M., Bémová, A., Hajič, J., Hajičová, E., Havelka, J., Kolářová, V., Kučová, L., Lopatková, M., Pajas, P., Panevová, J., et al. (2006). Annotation on the tectogrammatical layer in the prague dependency treebank, annotation manual.
- Miller, G. A. and Chomsky, N. (1963). Finitary models of language users. In *Handbook of Mathematical Psychology*.

- Myhill, J. (1985). Pragmatic and categorial correlates of vs word order. *Lingua*, 66:177–200.
- Neeleman, A. and van de Koot, H. (2016). Word order and information structure. In Féry, C. and Ishihara, S., editors, *The Oxford Handbook of Information Structure*.
- Newmeyer, F. J. (1992). Iconicity and generative grammar. *Language*, pages 756–796.
- Nicenboim, B. and Vasishth, S. (2018). Models of retrieval in sentence comprehension: A computational evaluation using bayesian hierarchical modeling. *Journal of Memory and Language*, 99:1–34.
- Nicenboim, B., Vasishth, S., Gattei, C., Sigman, M., and Kliegl, R. (2015). Working memory differences in long-distance dependency resolution. *Frontiers in Psychology*, 6:312.
- Nivre, J., Agić, Z., Ahrenberg, L., Antonsen, L., Aranzabe, M. J., Asahara, M., Ateyah, L., Attia, M., Atutxa, A., Augustinus, L., Badmaeva, E., Ballesteros, M., Banerjee, E., Bank, S., Barbu Mititelu, V., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bobicev, V., Börstell, C., Bosco, C., Bouma, G., Bowman, S., Burchardt, A., Candito, M., Caron, G., Cebiroğlu Eryiğit, G., Celano, G. G. A., Cetin, S., Chalub, F., Choi, J., Cinková, S., Çöltekin, C., Connor, M., Davidson, E., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dirix, P., Dobrovoljc, K., Dozat, T., Droganova, K., Dwivedi, P., Eli, M., Elkahky, A., Erjavec, T., Farkas, R., Fernandez Alcalde, H., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D., Garcia, M., Gärdenfors, M., Gerdes, K., Ginter, F., Goenaga, I., Gojenola, K., Gökirmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Gr̄uzītis, N., Guillaume, B., Habash, N., Hajič, J., Hajič jr., J., Hà My, L., Harris, K., Haug, D., Hladká, B., Hlaváčová, J., Hociung, F., Hohle, P., Ion, R., Irimia, E., Jelínek, T., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kayadelen, T., Kettnerová, V., Kirchner, J., Kotsyba, N., Krek, S., Laippala, V., Lambertino, L., Lando, T., Lee, J., Le Hong, P., Lenci, A., Lertpradit, S., Leung, H., Li, C. Y., Li, J., Li, K., Ljubešić, N., Loginova, O., Lyashevskaya, O., Lynn, T., Macketanz, V., Makazhanov, A., Mandl, M., Manning, C., Măranduc, C., Mareček, D., Marheinecke, K., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Mendonça, G., Miekka, N., Missilä, A., Mititelu, C., Miyao, Y., Montemagni, S., More, A., Moreno Romero, L., Mori, S., Moskalevskiy, B., Muischnek, K., Müürisep, K., Nainwani, P., Nedoluzhko, A., Nešpore-Bērzkalne, G., Nguyen Thi, L., Nguyen Thi Minh, H., Nikolaev, V., Nurmi, H., Ojala, S., Osenova, P., Östling, R., Øvreliid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Pitler, E., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Ramasamy, L., Rama, T., Ravishankar, V., Real, L., Reddy, S., Rehm, G., Rinaldi, L., Rituma, L., Romanenko, M., Rosa, R., Rovati, D., Sagot, B., Saleh, S., Samardžić, T., Sanguinetti, M., Saulite, B., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shen, M., Shimada, A., Sichinava, D., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Stella, A., Straka, M., Strnadová, J., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Trosterud, T., Trukhina, A., Tsarfaty, R., Tyers, F., Uematsu, S., Urešová, Z., Uria, L., Uszkoreit, H., Vajjala, S., van Niekerk, D., van Noord, G., Varga, V., Villemonte de la Clergerie, E., Vincze, V., Wallin, L., Washington, J. N., Wirén, M., Wong, T.-s., Yu, Z., Žabokrtský, Z., Zeldes, A., Zeman, D., and Zhu, H. (2017a). *Universal Dependencies 2.1*.
- Nivre, J., Agić, Ž., Ahrenberg, L., Antonsen, L., Aranzabe, M. J., Asahara, M., Ateyah, L., Attia, M., Atutxa, A., Badmaeva, E., Ballesteros, M., Banerjee, E., Bank, S., Bauer, J., Bengoetxea, K., Bhat, R. A., Bick, E., Bosco, C., Bouma, G., Bowman, S., Burchardt, A., Candito, M., Caron, G., Cebiroğlu Eryiğit, G., Celano, G. G. A., Cetin, S., Chalub, F., Choi, J., Cho, Y., Cinková, S., Çöltekin, Ç., Connor, M., de Marneffe, M.-C., de Paiva, V., Diaz de Ilarraza, A., Dobrovoljc, K., Dozat, T., Droganova, K., Eli, M., Elkahky, A., Erjavec, T., Farkas, R., Fernandez Alcalde, H., Foster, J., Freitas, C., Gajdošová, K., Galbraith, D.,

- Garcia, M., Ginter, F., Goenaga, I., Gojenola, K., Gökırmak, M., Goldberg, Y., Gómez Guinovart, X., Gonzáles Saavedra, B., Grioni, M., Grūzītis, N., Guillaume, B., Habash, N., Hajič, J., Hajič jr., J., Hà Mỷ, L., Harris, K., Haug, D., Hladká, B., Hlaváčová, J., Hohle, P., Ion, R., Irimia, E., Johannsen, A., Jørgensen, F., Kaşıkara, H., Kanayama, H., Kanerva, J., Kayadelen, T., Kettnerová, V., Kirchner, J., Kotsyba, N., Krek, S., Kwak, S., Laippala, V., Lambertino, L., Lando, T., Lê H'ông, P., Lenci, A., Lertpradit, S., Leung, H., Li, C. Y., Li, J., Ljubešić, N., Loginova, O., Lyashevskaya, O., Lynn, T., Macketanz, V., Makazhanov, A., Mandl, M., Manning, C., Manurung, R., Măranduc, C., Mareček, D., Marheinecke, K., Martínez Alonso, H., Martins, A., Mašek, J., Matsumoto, Y., McDonald, R., Mendonça, G., Missilä, A., Mititelu, V., Miyao, Y., Montemagni, S., More, A., Moreno Romero, L., Mori, S., Moskalevskyi, B., Muischnek, K., Mustafina, N., Müürisep, K., Nainwani, P., Nedoluzhko, A., Nguyễn Thị, L., Nguyễn Thị Minh, H., Nikolaev, V., Nitisaraj, R., Nurmi, H., Ojala, S., Osenova, P., Øvrelid, L., Pascual, E., Passarotti, M., Perez, C.-A., Perrier, G., Petrov, S., Piitulainen, J., Pitler, E., Plank, B., Popel, M., Pretkalniņa, L., Prokopidis, P., Puolakainen, T., Pyysalo, S., Rademaker, A., Real, L., Reddy, S., Rehm, G., Rinaldi, L., Rituma, L., Rosa, R., Rovati, D., Saleh, S., Sanguinetti, M., Saulīte, B., Sawanakunanon, Y., Schuster, S., Seddah, D., Seeker, W., Seraji, M., Shakurova, L., Shen, M., Shimada, A., Shohibussirri, M., Silveira, N., Simi, M., Simionescu, R., Simkó, K., Šimková, M., Simov, K., Smith, A., Stella, A., Strnadová, J., Suhr, A., Sulubacak, U., Szántó, Z., Taji, D., Tanaka, T., Trosterud, T., Trukhina, A., Tsarfaty, R., Tyers, F., Uematsu, S., Urešová, Z., Uria, L., Uszkoreit, H., van Noord, G., Varga, V., Vincze, V., Washington, J. N., Yu, Z., Žabokrtský, Z., Zeman, D., and Zhu, H. (2017b). Universal dependencies 2.0 – CoNLL 2017 shared task development and test data. LINDAT/CLARIN digital library at the Institute of Formal and Applied Linguistics (ÚFAL), Faculty of Mathematics and Physics, Charles University.
- Rambow, O. and Joshi, A. K. (2015). 12 a processing model for free word-order languages. *Perspectives on sentence processing*.
- Rasmussen, N. E. and Schuler, W. (2018). Left-corner parsing with distributed associative memory produces surprisal and locality effects. *Cognitive science*, 42:1009–1042.
- Resnik, P. (1992). Left-corner parsing and psychological plausibility. In *Proceedings of the 14th conference on Computational linguistics-Volume 1*, pages 191–197. Association for Computational Linguistics.
- Rijkhoff, J. (1990). Explaining word order in the noun phrase. *Linguistics*, 28(1):5–42.
- Ross, J. R. (1967). Constraints on variables in syntax. Dissertation.
- Schach, S., Gottwald, S., and Braun, D. A. (2018). Quantifying motor task performance by bounded rational decision theory. *Frontiers in neuroscience*, 12:932.
- Shalizi, C. R. and Crutchfield, J. P. (2001). Computational mechanics: Pattern and prediction, structure and simplicity. *Journal of statistical physics*, 104(3-4):817–879.
- Shannon, C. E. (1951). Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64.
- Sharan, V., Kakade, S., Liang, P., and Valiant, G. (2016). Prediction with a Short Memory. *arXiv:1612.02526 [cs, stat]*. arXiv: 1612.02526.
- Sims, C. R. (2018). Efficient coding explains the universal law of generalization in human perception. *Science*, 360(6389):652–656.

- Sims, C. R., Jacobs, R. A., and Knill, D. C. (2012). An ideal observer analysis of visual working memory. *Psychological review*, 119(4):807.
- Smith, N. J. and Levy, R. (2013). The effect of word predictability on reading time is logarithmic. *Cognition*, 128(3):302–319.
- Stallings, L. M. and MacDonald, M. C. (2011). It’s not just the “heavy np”: relative phrase length modulates the production of heavy-np shift. *Journal of psycholinguistic research*, 40(3):177–187.
- Staub, A. and Clifton Jr, C. (2006). Syntactic prediction in language comprehension: Evidence from either... or. *Journal of experimental psychology: Learning, memory, and cognition*, 32(2):425.
- Staub, A., Clifton Jr, C., and Frazier, L. (2006). Heavy np shift is the parser’s last resort: Evidence from eye movements. *Journal of memory and language*, 54(3):389–406.
- Still, S. (2014). Information Bottleneck Approach to Predictive Inference. *Entropy*, 16(2):968–989.
- Still, S., Sivak, D. A., Bell, A. J., and Crooks, G. E. (2012). Thermodynamics of prediction. *Physical review letters*, 109(12):120604.
- Takahashi, S. and Tanaka-Ishii, K. (2018). Cross entropy of neural language models at infinity—a new bound of the entropy rate. *Entropy*, 20(11):839.
- Tanaka, T., Miyao, Y., Asahara, M., Uematsu, S., Kanayama, H., Mori, S., and Matsumoto, Y. (2016). Universal dependencies for japanese. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC’16)*, pages 1651–1658.
- Temperley, D. and Gildea, D. (2018). Minimizing Syntactic Dependency Lengths: Typological/Cognitive Universal? *Annu. Rev. Linguist*, 4:1–15.
- Tesnière, L. and Kahane, S. (2015). *Elements of structural syntax*. John Benjamins Publishing Company New York.
- Vasishth, S., Nicenboim, B., Engelmann, F., and Burchert, F. (2019). Computational models of retrieval processes in sentence processing. *Trends in Cognitive Sciences*.
- Vasishth, S., Suckow, K., Lewis, R. L., and Kern, S. (2010). Short-term forgetting in sentence comprehension: Crosslinguistic evidence from verb-final structures. *Language and Cognitive Processes*, 25(4):533–567.
- Wang, D. and Eisner, J. (2016). The Galactic Dependencies Treebanks: Getting more data by synthesizing new languages. *Transactions of the Association for Computational Linguistics*, 4:491–505.
- Wells, J. B., Christiansen, M. H., Race, D. S., Acheson, D. J., and MacDonald, M. C. (2009). Experience and sentence processing: Statistical learning and relative clause comprehension. *Cognitive psychology*, 58(2):250–271.
- Yngve, V. H. (1960). A model and an hypothesis for language structure. *Proceedings of the American philosophical society*, 104(5):444–466.
- Yuret, D. (1998). Discovery of linguistic relations using lexical attraction. *arXiv preprint cmp-lg/9805009*.

- Zaslavsky, N., Kemp, C., Regier, T., and Tishby, N. (2018). Efficient compression in color naming and its evolution. *Proceedings of the National Academy of Sciences*, 115(31):7937–7942.
- Zénon, A., Solopchuk, O., and Pezzulo, G. (2019). An information-theoretic perspective on the costs of cognition. *Neuropsychologia*, 123:5–18.

Supplementary Information for: Modeling word and morpheme order in natural language as an efficient tradeoff of memory and surprisal

Michael Hahn, Judith Degen, Richard Futrell

June 5, 2020

Contents

1	Formal Analysis and Proofs	2
1.1	Mathematical Assumptions	2
1.1.1	Ingredient 1: Language as a Stationary Stochastic Process	2
1.1.2	Ingredient 2: Postulates about Memory and Processing	3
1.1.3	Ingredient 3: No Mindreading	4
1.2	Proof of the Theorem	4
1.3	Memory-Surprisal Tradeoff in a Model with Memory Retrieval	5
1.4	Information Locality in Language Production	8
1.4.1	Information Locality Theorem in Production	10
1.5	Proof of Left-Right Invariance	11
2	Examples with Analytical Calculations	11
2.1	Window-Based Model not Optimal	12
2.2	Tight Bound for Retrieval Model	12
2.3	Low memory requirements do not imply decay of unconditional mutual information	14
3	Study 2	14
3.1	Corpus Size per Language	14
3.2	Details for Neural Network Models	15
3.2.1	Choice of Hyperparameters	15
3.2.2	Estimation of average surprisal	16
3.2.3	Number of Samples, Precision-Based Stopping Criterion	17
3.3	Samples Drawn per Language	17
3.4	N-Gram Models	19
3.4.1	Method	19
3.4.2	Results	19
3.5	Chart Parsing Control	23
3.5.1	Deriving PCFGs from Dependency Corpora	23
3.5.2	Estimating I_t with Chart Parsing	24
3.5.3	Results	25

4 Study 3	27
4.1 Determining Japanese Verb Suffixes	27
4.2 Determining Sesotho Verb Affixes	28
4.3 Experiment	31

1 Formal Analysis and Proofs

In this section, we prove the Information Locality Bound Theorem and related theoretical results referenced in the main paper.

1.1 Mathematical Assumptions

We first make explicit how we formalize language processing for proving the theorem. This is a formally fully rigorous statement of the model described in main paper (Section ‘An information-theoretic model of online language comprehension’).

1.1.1 Ingredient 1: Language as a Stationary Stochastic Process

We represent language as a stochastic process of words $W = \dots w_{-2}w_{-1}w_0w_1w_2\dots$, extending indefinitely both into the past and into the future (Doob, 1953). The symbols w_t belong to a common set, representing the words or morphemes of the language. Formally, a stochastic process is a probability distribution over infinite sequences $\dots w_{-2}w_{-1}w_0w_1w_2\dots$ (Doob, 1953). As t runs over the set of integers \mathbb{Z} , it will sometimes be convenient to write such an infinite sequence as $(w_t)_{t \in \mathbb{Z}}$. This distribution gives rise to probability distributions over finite subsequences

$$P(w_t, \dots, w_{t+T}) \tag{1}$$

for integers t, T , and to conditional probabilities

$$P(w_t | w_{t-T}, \dots, w_{t-1}) \tag{2}$$

Infinite Length We assume that the process W extends infinitely into both past and future, whereas real words, sentences, and conversations are finite. This is not a contradiction: In Studies 1–3, we model W as a sequence of independent sentences or words, separated with a special “end-of-sentence” symbol. Modeling W as such an infinite sequence of finite sentences provides a way to formalize the memory-surprisal tradeoff in a way independent of the time point t .

Stationarity We make the assumption that the process W is *stationary* (Doob, 1953). This means that the joint distribution of different symbols depends only on their *relative positions*, not their *absolute positions*. Formally, this means that joint probabilities do not change when shifting all observations by a constant number Δ of time steps. That is, for any integers t, Δ , and $T > 0$:

$$P(w_t, \dots, w_{t+T}) = P(w_{t+\Delta}, \dots, w_{t+T+\Delta}) \tag{3}$$

Informally, this says that the process has no ‘internal clock’, and that the statistical rules of the language do not change over time at the timescale we are interested in. In reality, the statistical rules of language do change: They change as language changes over generations, and they also change between different

situations – e.g., depending on the interlocutor at a given point in time. However, we are interested in memory needs in the processing of *individual sentences* or *individual words*, at a timescale of seconds or minutes. At this level, the statistical regularities of language do not change, making stationarity a reasonable modeling assumption.

The choice to model language as a stationary stochastic process is common to information-theoretic studies of text, including studies of entropy rate (Shannon, 1951; Bentz et al., 2017; Takahashi and Tanaka-Ishii, 2018), excess entropy (Debowski, 2011; Hahn and Futrell, 2019), and mutual information (Ebeling and Pöschel, 1994; Lin and Tegmark, 2017).

1.1.2 Ingredient 2: Postulates about Memory and Processing

The second ingredient consists of the three postulates about memory and processing described in the main paper. We repeat these here for reference:

1. Comprehension Postulate 1 (Incremental memory). At time t , the listener has an incremental **memory state** m_t that contains her stored information about previous words. The memory state is given by a **memory encoding function** M such that $m_t = M(w_{t-1}, m_{t-1})$.
2. Comprehension Postulate 2 (Incremental prediction). The listener has a subjective probability distribution at time t over the next word w_t as a function of the memory state m_t . This probability distribution is denoted $P(w_t|m_t)$.
3. Comprehension Postulate 3 (Linking hypothesis). Processing a word w_t incurs difficulty proportional to the **surprisal** of w_t given the memory state m_t :

$$\text{Difficulty} \propto -\log P(w_t|m_t). \quad (4)$$

We extend the assumption of stationarity explained above to the memory state m_t , modeling the pair $(w_t, m_t)_{t \in \mathbb{Z}}$ as a stationary process. Formally, this means that, for any integers t , Δ , and $T > 0$:

$$P((w_t, m_t), \dots, (w_{t+T}, m_{t+T})) = P((w_{t+\Delta}, m_{t+\Delta}), \dots, (w_{t+T+\Delta}, m_{t+T+\Delta})) \quad (5)$$

This means that the listener’s memory state only depends on the relative temporal position of past observed symbols, not on any absolute time scale. This prevents situations where the listener’s memory state keeps track of some absolute notion of time (e.g., counting whether t is even or odd) even though the statistical regularities of the input $(w_t)_{t \in \mathbb{Z}}$ are independent of time.

This assumption entails that average surprisal

$$S_M \equiv H[w_t|m_t]. \quad (6)$$

and memory cost

$$H_M \equiv H[m_t] \quad (7)$$

are independent of t , as these terms only depend on the joint distribution of (w_t, m_t) , which is independent of t .

1.1.3 Ingredient 3: No Mindreading

Our postulates so far do not rule out that the listener has access to information that was never revealed during past interaction. That is, they permit situations where m_t maintains some information that is not contained in the past inputs $w_{<t} = (\dots, w_{t-2}, w_{t-1})$, but is informative about future input $w_{\geq t} = (w_t, w_{t+1}, w_{t+2}, \dots)$. Such a situation would correspond to a listener ‘mindreading’ the speaker’s intentions. We exclude this by explicitly stating that the listener has no access to information about the future beyond what is contained in the past. We formalize this as saying that the memory state is independent of future observations, conditional on the past:

$$m_t \perp w_{\geq t} | w_{<t} \quad (8)$$

Remarkably, the Information Locality Theorem can be proved even without this assumption. However, this assumption is necessary in order to prove that $S_M \geq S_\infty$ even for very large memory capacities, i.e., that imperfect memory can never lead to lower average surprisal than the entropy rate. Such a situation could only be achieved if the listener somehow ‘read the speaker’s mind’.

There are no further assumptions about the memory architecture and the nature of its computations.

1.2 Proof of the Theorem

Here, we prove the Information Locality Bound Theorem (Theorem 2 in the main paper) based on the assumptions described in the previous section. Recall that S_M and S_∞ are given by

$$S_M \equiv H[w_t | m_t] \quad (9)$$

$$S_\infty \equiv H[w_t | w_{<t}] \quad (10)$$

We restate the theorem:

Theorem 1. *Let T be any positive integer ($T \in \{1, 2, 3, \dots\}$), and consider a listener using at most*

$$H_M \leq \sum_{t=1}^T t I_t \quad (11)$$

bits of memory on average. Then this listener will incur surprisal at least

$$S_M \geq S_\infty + \sum_{t>T} I_t \quad (12)$$

on average.

Proof. The difference between the listener’s average surprisal S_M and optimal surprisal S_∞ is

$$S_M - S_\infty = H[w_t | m_t] - H[w_t | w_{<t}]. \quad (13)$$

Because the process $(w_t, m_t)_{t \in \mathbb{Z}}$ is stationary, we can, for any positive integer T , rewrite this expression as

$$H[w_t | m_t] - H[w_t | w_{<t}] = \frac{1}{T} \sum_{t'=1}^T (H[w_{t'} | m_{t'}] - H[w_{t'} | w_{<t'}]) \quad (14)$$

Due to Processing Postulate 1, we have

$$m_t = M(m_{t-1}, w_{t-1}) = M(M(m_{t-2}, w_{t-2}), w_{t-1}) = M(M(M(m_{t-3}, w_{t-3}), w_{t-2}), w_{t-1}) = \dots, \quad (15)$$

and therefore the Data Processing Inequality (Cover and Thomas, 2006) entails the following inequality for every positive integer t :

$$H[w_t|m_t] \geq H[w_t|w_{1..t-1}, m_1]. \quad (16)$$

Plugging this inequality into Equation 14 above, we get an expression in terms of the difference in mutual information between a block of words and a memory representation, and a block of words and the true past:

$$H[w_t|m_t] - H[w_t|w_{<t}] \geq \frac{1}{T} \sum_{t=1}^T (H[w_t|w_{1..t-1}, m_1] - H[w_t|w_{1..t-1}, w_{\leq 0}]) \quad (17)$$

$$= \frac{1}{T} (H[w_{1..T}|m_1] - H[w_{1..T}|w_{\leq 0}]) \quad (18)$$

$$= \frac{1}{T} (I[w_{1..T} : w_{\leq 0}] - I[w_{1..T} : m_1]). \quad (19)$$

The first term $I[w_{1..T} : w_{\leq 0}]$ can be rewritten in terms of I_t using the chain rule of mutual information (Cover and Thomas, 2006):

$$I[w_{1..T} : w_{\leq 0}] = \sum_{i=1}^T \sum_{j=-\infty}^{-1} I[w_i : w_j | w_{j+1} \dots w_{i-1}] = \sum_{t=1}^T t I_t + T \sum_{t>T} I_t. \quad (20)$$

Therefore

$$H[w_t|m_t] - H[w_t|w_{<t}] \geq \frac{1}{T} \left(\sum_{t=1}^T t I_t + T \sum_{t>T} I_t - I[w_{1..T} : m_1] \right). \quad (21)$$

The term $I[w_{1..T} : m_1]$ is at most $H[m_1]$, which is at most $\sum_{t=1}^T t I_t$ by assumption. Thus, (21) implies the following:

$$H[w_t|m_t] - H[w_t|w_{<t}] \geq \frac{1}{T} \left(\sum_{t=1}^T t I_t + T \sum_{t>T} I_t - \sum_{t=1}^T t I_t \right) = \sum_{t>T} I_t \quad (22)$$

Rearranging yields

$$H[w_t|m_t] \geq H[w_t|w_{<t}] + \sum_{t>T} I_t \quad (23)$$

as claimed. \square

1.3 Memory-Surprisal Tradeoff in a Model with Memory Retrieval

Here we show that our information-theoretic analysis is compatible with models placing the main bottleneck in the difficulty of retrieval (McElree, 2000; Lewis and Vasishth, 2005; Nicenboim and Vasishth, 2018; Vasishth et al., 2019). We extend our model of memory in incremental prediction to capture key aspects of the models described by Lewis and Vasishth (2005); Nicenboim and Vasishth (2018); Vasishth et al. (2019).

The ACT-R model of Lewis and Vasishth (2005) assumes a small working memory consisting of *buffers* and a *control state*, which together hold a small and fixed number of individual *chunks*. It also assumes a large short-term memory that contains an unbounded number of chunks. This large memory store is accessed via *cue-based retrieval*: a query is constructed based on the current state of the buffers and the control state; a chunk that matches this query is then selected from the memory storage and placed into one of the buffers.

Formal Model We extend our information-theoretic analysis by considering a model that maintains both a small working memory m_t —corresponding to the buffers and the control state—and an unlimited short-term memory s_t . When processing a word w_t , there is some amount of communication between m_t and s_t , corresponding to retrieval operations. We model this using a variable r_t representing the information that is retrieved from s_t . In our formalization, r_t reflects the totality of all retrieval operations that are made during the processing of w_{t-1} ; they happen after w_{t-1} has been observed but before w_t has.

The working memory state is determined not just by the input w_t and the previous working memory state m_{t-1} , but also by the retrieved information:

$$m_t = f(w_t, m_{t-1}, r_t) \quad (24)$$

The retrieval operation is jointly determined by working memory, short-term memory, and the previous word:

$$r_t = g(w_{t-1}, m_{t-1}, s_{t-1}) \quad (25)$$

Finally, the short-term memory can incorporate any—possibly all—information from the last word and the working memory:

$$s_t = h(w_t, m_t, s_{t-1}) \quad (26)$$

While s_t is unconstrained, there are constraints on the capacity of working memory $H[m_t]$ and the amount of retrieved information $H[r_t]$. Placing a bound on $H[m_t]$ reflects the fact that the buffers can only hold a small and fixed number of chunks (Lewis and Vasishth, 2005).

Predictions are made based on working memory m_{t-1} and retrieved information r_t (but not the short-term memory s_t), incurring average surprisal

$$S := H[w_t | m_{t-1}, r_t]. \quad (27)$$

In line with the mathematical postulates in Section 1.1, we assume that $(w_t, m_t, r_t, s_t)_{t \in \mathbb{Z}}$ is stationary as a stochastic process.

Cost of Retrieval In the model of Lewis and Vasishth (2005), the time it takes to process a word is determined primarily by the time spent retrieving chunks, which is determined by the number of retrieval operations and the time it takes to complete each retrieval operation. If the information content of each chunk is bounded, then a bound on $H[r_t]$ corresponds to a bound on the number of retrieval operations.

In the model of Lewis and Vasishth (2005), a retrieval operation takes longer if more chunks are similar to the retrieval cue, whereas, in the direct-access model (McElree, 2000; Nicenboim and Vasishth, 2018; Vasishth et al., 2019), retrieval operations take a constant amount of time. There is no direct counterpart to differences in retrieval times and similarity-based inhibition as in the activation-based model in our formalization. Our formalization thus more closely matches the direct-access model, though it might be possible to incorporate aspects of the activation-based model in our formalization.

Role of Surprisal The ACT-R model of Lewis and Vasishth (2005) does not have an explicit surprisal cost. Instead, surprisal effects are interpreted as arising because, in less constraining contexts, the parser is more likely to make decisions that then turn out to be incorrect, leading to additional correcting steps. We view this as an algorithmic-level implementation of a surprisal cost. If the word w_t is unexpected given the current state of the working memory—i.e., buffers and control states—then their current state must provide insufficient information to constrain the actual syntactic state of the sentence, meaning that the parsing steps

made to integrate w_t are likely to include more backtracking and correction steps. Thus, we argue that cue-based retrieval models predict that the surprisal $-\log P(w_t|m_{t-1}, r_t)$ will be part of the cost of processing word w_t .

Theoretical Result We now show an extension of our theoretical result in the setting of the retrieval-based model described above.

Theorem 2. *Let $0 < S \leq T$ be positive integers such that the average working memory cost $H[m_t]$ is bounded as*

$$H[m_t] \leq \sum_{t=1}^T tI_t \quad (28)$$

and the average amount of retrieved information is bounded as

$$H[r_t] \leq \sum_{t=T+1}^S I_t. \quad (29)$$

Then the surprisal cost is lower-bounded as

$$H[w_t|m_{t-1}, r_t] \geq H[w_t|w_{<t}] + \sum_{t>S} I_t. \quad (30)$$

Proof. The proof is a generalization of the proof in Section 1.2. For any positive integer t , the memory state m_t is determined by $w_{1...t}, m_0, r_0, \dots, r_t$. Therefore, the Data Processing Inequality entails:

$$H[w_t|m_{t-1}, r_t] \geq H[w_t|w_{1...t}, m_0, r_0, \dots, r_t]. \quad (31)$$

As in (17), this leads to

$$H[w_t|m_{t-1}, r_t] - H[w_t|w_{<t}] \geq \frac{1}{T} \sum_{t=1}^T (H[w_t|w_{1...t}, m_0, r_0, \dots, r_t] - H[w_t|w_{1...t-1}, w_{\leq 0}]) \quad (32)$$

$$\geq \frac{1}{T} (H[w_{1...T}|m_0, r_0, \dots, r_T] - H[w_{1...T}|w_{\leq 0}]) \quad (33)$$

$$= \frac{1}{T} (I[w_{1...T}, w_{\leq 0}] - I[w_{1...T}, (m_0, r_0, \dots, r_T)]). \quad (34)$$

Now, using the calculation from (20), this can be rewritten as:

$$\begin{aligned} H[w_t|m_{t-1}, r_t] - H[w_t|w_{<t}] &= \frac{1}{T} \left(\sum_{t=1}^T tI_t + T \sum_{t>T} I_t - I[w_1 \dots w_T, (m_0, r_1, \dots, r_T)] \right) \\ &= \frac{1}{T} \left(\sum_{t=1}^T tI_t + T \sum_{t>T} I_t - I[w_{1...T}, m_0] - \sum_{t=1}^T I[w_{1...T}, r_t|m_0, r_{1...t-1}] \right). \end{aligned}$$

Due to the inequalities

$$I[w_{1...T}, m_0] \leq H[m_0] \leq \sum_{t=1}^T tI_t \quad (35)$$

$$I[w_{1...T}, r_t|m_0, r_{1...t-1}] \leq H[r_t] \leq \sum_{t=T+1}^S I_t, \quad (36)$$

this can be bounded as

$$H[w_t|m_{t-1}, r_t] - H[w_t|w_{<t}] \geq \frac{1}{T} \left(\sum_{t=1}^T tI_t + T \sum_{t>T} I_t - H[m_0] - \sum_{t=1}^T H[r_t] \right). \quad (37)$$

Finally, this reduces as

$$H[w_t|m_{t-1}, r_t] - H[w_t|w_{<t}] \geq \frac{1}{T} (T \sum_{t>T} I_t - T \cdot H[r_t]) \quad (38)$$

$$= \sum_{t>T} I_t - H[r_t] \quad (39)$$

$$\geq \sum_{t>T} I_t - \sum_{t=T+1}^S I_t \quad (40)$$

$$= \sum_{t>S} I_t. \quad (41)$$

□

Information Locality We now show that this result predicts information locality provided that retrieving information is more expensive than keeping the same amount of information in working memory. For this, we formalize the problem of finding an optimal memory strategy as a multi-objective optimization, aiming to minimize

$$\lambda_1 H[m_t] + \lambda_2 H[r_t]. \quad (42)$$

to achieve a given surprisal level, for some setting of $\lambda_1, \lambda_2 > 0$ describing the relative cost of storage and retrieval. What is the optimal division of labor between keeping information in working memory and recovering it through retrieval? The problem

$$\min_T \lambda_1 \sum_{t=1}^T tI_t + \lambda_2 \sum_{t=T+1}^S I_t \quad (43)$$

has solution $T \approx \frac{\lambda_2}{\lambda_1}$. This means that, as long as retrievals are more expensive than keeping the same amount of information in working memory (i.e., $\lambda_2 > \lambda_1$), the optimal strategy stores information from the last $T > 1$ words in working memory. Due to the factor t inside $\sum_{t=1}^T tI_t$, the bound (43) will be reduced when I_t decays faster, i.e., there is strong information locality.

The assumption that retrieving information is more difficult than storing it is reasonable for cue-based retrieval models, as retrieval suffers from similarity-based interference effects due to the unstructured nature of the storage (Lewis and Vasishth, 2005). A model that maintains no information in its working memory, i.e. $H[m_t] = 0$, would correspond to a cue-based retrieval model that stores nothing in its buffers and control states, and relies entirely on retrieval to access past information. Given the nature of representations assumed in models (Lewis and Vasishth, 2005), such a model would seem to be severely restricted in its ability to parse language.

1.4 Information Locality in Language Production

Here we show results linking memory and locality in production. We show that results similar to our main theorem hold for the tradeoff between a speaker's memory and the accuracy with which they match the distribution of the language.

In the case of production, the memory–surprisal trade-off arises from the minimization of error in production of linguistic sequences. That is, given a **competence language** (a target distribution on words given contexts), a speaker tries to produce a **performance language** which is as close as possible to the competence language. The performance language operates under memory constraints, so the performance language will diverge from the competence language due to production errors. When a speaker has more incremental memory about what she has already produced, then she is able to produce linguistic sequences with less error, thus reducing the divergence between the performance language and the competence language. The reduction of this competence–performance divergence for a speaker is formally equivalent to the minimization of average surprisal for a listener.

Formally, we assign a speaker a production policy $q(w_t|m_t)$ that produces the next word conditional on the speaker’s memory state m_t . We assume that speakers aim to minimize the occurrence of production errors. We formalize this as minimizing the KL divergence from the performance language $q(w_t|m_t)$ to the target competence language $p(w_t|w_{<t})$. We call this divergence the **competence–performance divergence** under the memory encoding function M and the production policy q :

$$d_M^q \equiv D_{\text{KL}}[p(w_t|w_{<t})||q(w_t|m_t)] \quad (44)$$

$$= \sum_{w_{\leq t}} p(w_{\leq t}) \log \frac{p(w_t|w_{<t})}{q(w_t|m_t)}. \quad (45)$$

Under this assumption, the Information Locality Bound Theorem will apply in production as well as comprehension: The competence–performance divergence d_M^q trades off with memory load $H[m_t]$, and this tradeoff will be more favorable when languages exhibit information locality. This means that languages that exhibit information locality can be produced with greater accuracy given limited memory resources.

We derive the existence of this trade-off from the following postulates about language production. Let the competence language be represented by a stationary stochastic process, parameterized by a probability distribution $p(w_t|w_{<t})$ giving the conditional probability of any word w_t given an unbounded number of previous words. Our postulates describe a speaker who tries to find a performance language $q(w_t|m_t)$ to match the the competence language using incremental memory representations m_t :

1. **Production Postulate 1 (Incremental memory).** At time t , the speaker has an incremental **memory state** m_t that contains (1) her stored information about previous words that she has produced, and (2) information about her production target. The memory state is given by a **memory encoding function** M such that $m_t = M(w_{t-1}, m_{t-1})$.
2. **Production Postulate 2 (Production policy).** At time t , the speaker produces the next word w_t conditional on her memory state by drawing from a probability distribution $q(w_t|m_t)$. We call q the speaker’s **production policy**.
3. **Production Postulate 3 (Minimizing divergence).** The production policy q is selected to minimize the KL divergence from the performance language to the target competence language $p(w_t|w_{<t})$. We call this divergence the **competence–performance divergence** under the memory encoding function M and the production policy q :

$$d_M^q \equiv D_{\text{KL}}[p(w_t|w_{<t})||q(w_t|m_t)] \quad (46)$$

$$= \sum_{w_{\leq t}} p(w_{\leq t}) \log \frac{p(w_t|w_{<t})}{q(w_t|m_t)}. \quad (47)$$

Completing the link with the memory–surprisal trade-off in comprehension, we note that when the production policy $q(w_t|m_t)$ is selected to minimize the competence–performance divergence d_M^q , then this divergence becomes equal to the memory distortion $S_M - S_\infty$ discussed in the context of comprehension costs. Therefore, under these postulates, the Information Locality Bound Theorem will apply in production as well as comprehension (see Section 1.4.1 for formal statement and proof). This means that languages that exhibit information locality can be produced with greater accuracy given limited memory resources.

In the case of language comprehension, the trade-off represented excess processing *difficulty* arising due to memory constraints. In the case of language production, the trade-off represents *production error* arising due to memory constraints. When memory is constrained, then the speaker’s productions will diverge from her target language. And as memory is more and more constrained, this divergence will increase more and more. The degree of divergence is measured in the same units as surprisal, hence the formal equivalence between the listener’s and speaker’s memory–surprisal trade-offs.

Although the memory–surprisal trade-off is mathematically similar between comprehension and production, it is not necessarily identical. The comprehender’s memory–surprisal trade-off has to do with the amount of predictive information I_t stored in memory, where I_t is defined in terms of a probability distribution on words given t words of context. In the producer’s memory–surprisal tradeoff, this probability distribution may be different, because the producer has knowledge of a production target (Production Postulate 1). Nevertheless, if the producer’s probability distribution is similar to the comprehender’s, then we predict the same trade-off for the producer as for the comprehender.

It may be possible to use this asymmetry to distinguish whether word and morpheme order is more optimized for the comprehender or the producer. If word order is best predicted under a probability model that uses zero information about a production target (as in the current work), then we have evidence that the comprehender’s trade-off is more important. On the other hand, if word order is best predicted under a probability model that uses (partial) information about a production target, then we have evidence that the producer’s trade-off is more important. As estimating the difference between these probability distributions is difficult, we leave this avenue of research to future work.

1.4.1 Information Locality Theorem in Production

Here, we prove an Information Locality Theorem in production. Following the Production Postulates 1–3, we consider a setting in which a speaker produces sentences with bounded memory, and analyze the deviation of the produced distribution from the actual distribution of the language. We consider a speaker who maintains memory representations and incrementally produces based on these representations:

$$P_{\text{produced}}(w_t|w_{<t}) = q(w_t|m_t) \quad (48)$$

We show a tradeoff between the memory capacity $H[m_t]$ and the KL-divergence between the actual language statistics and the speaker’s production distribution, as defined in Production Postulate 3:

$$d_M^q = D_{KL}(P_{\text{language}}||P_{\text{produced}}) = \mathbb{E}_{w_{<t}} \sum_{w_t} p(w_t|w_{<t}) \log \frac{p(w_t|w_{<t})}{p_{\text{produced}}(w_t|w_{<t})} \quad (49)$$

As in the case of comprehension, we model $(w_t, m_t)_{t \in \mathbb{Z}}$ as stationary; however, we do *not* assume the ‘No Mindreading’ condition (8).

Theorem 3. *If a speaker maintains memory*

$$H[m_t] \leq \sum_{i=1}^T tI_t, \quad (50)$$

then

$$d_M^q = D_{KL}(P_{\text{language}} || P_{\text{produced}}) \geq \sum_{t=T+1}^{\infty} I_t. \quad (51)$$

While this bound only considers the production of a single word, it entails a bound on the production accuracy for sequences:

$$D_{KL}(P_{\text{language}}(w_1 \dots w_t | w_{\leq 0}) || P_{\text{produced}}(w_1 \dots w_t | w_{\leq 0})) = t \cdot D_{KL}(P_{\text{language}}(w_1 | w_{\leq 0}) || P_{\text{produced}}(w_1 | w_{\leq 0})) \quad (52)$$

Proof. We rewrite the KL-Divergence so that we can reduce this result to the proof in the comprehension setting (Section 1.2). First note

$$D_{KL}(P_{\text{language}} || P_{\text{produced}}) = \mathbb{E}_{w_{<t}} \left[\sum_{w_t} p(w_t | w_{<t}) \log \frac{p(w_t | w_{<t})}{p_{\text{produced}}(w_t | w_{<t})} \right] \quad (53)$$

$$= \mathbb{E}_{w_{<t}} \left[\sum_{w_t} p(w_t | w_{<t}) \log \frac{p(w_t | w_{<t})}{p(w_t | M(w_{<t}))} \right] \quad (54)$$

$$= \mathbb{E}_{w_{<t}} \left[\sum_{w_t} p(w_t | w_{<t}) \log p(w_t | w_{<t}) \right] - \mathbb{E}_{w_{<t}} \left[\sum_{w_t} p(w_t | w_{<t}) \log p(w_t | M(w_{<t})) \right] \quad (55)$$

$$= H[w_t | M(w_{<t})] - H[w_t | w_{<t}] \quad (56)$$

We now note that the proof in Section 1.2 can be used, without further modification, to show that

$$H[w_t | M(w_{<t})] - H[w_t | w_{<t}] \geq \sum_{t=T+1}^{\infty} I_t \quad (57)$$

completing the proof. The reason we can apply the proof from Section 1.2 is that Comprehension Postulate 1, where it is used in that proof, can be replaced by the analogous Production Postulate 1. \square

1.5 Proof of Left-Right Invariance

Here we show that the bound provided by the Information Locality Theorem is invariant under reversal of the process. That is: Given a process $(X_t)_{t \in \mathbb{Z}}$, we define its reverse process $(Y_t)_{t \in \mathbb{Z}}$ by $Y_t := X_{-t}$. We claim that the theorem provides the same bounds for the memory-surprisal tradeoff curves. To prove this, we note:

$$I[X_t, X_0 | X_{1 \dots t-1}] = I[Y_{-t}, Y_0 | Y_{1-t \dots -1}] = I[Y_0, Y_t | Y_{1 \dots t-1}] = I[Y_t, Y_0 | Y_{1 \dots t-1}] \quad (58)$$

The first step follows from the definition of Y . The second step follows from the fact that X_t , and thus also Y_t , is stationary, and thus adding t to each index in the expression does not change the resulting value. The third step uses the fact that mutual information is symmetric.

2 Examples with Analytical Calculations

Here, we provide examples of the Information Locality Theorem in settings where analytical calculations are possible. These examples are artificial and intended to demonstrate the mathematical possibility of certain phenomena; we do not intend these examples to model any linguistic phenomena.

2.1 Window-Based Model not Optimal

Here we provide an example of a stochastic process where a window-based memory encoding is not optimal, but the bound provided by our theorem still holds. This is an example where the bound provided by the theorem is not tight: while it bounds the memory-surprisal tradeoff of all possible listeners, the bound is ‘optimistic’, meaning that no mathematically possible memory encoding function M can exactly achieve the bound.

Let k be some positive integer. Consider a process $x_{t+1} = (v_{t+1}, w_{t+1}, y_{t+1}, z_{t+1})$ where

1. The first two components consist of fresh random bits. Formally, v_{t+1} is an independent draw from $Bernoulli(0.5)$, independent from all preceding observations $x_{\leq t}$. Second, let w_{t+1} consist of $2k$ many such independent random bits (so that $H[w_{t+1}] = 2k$)
2. The third component *deterministically* copies the first bit from $2k$ steps earlier. Formally, y_{t+1} is equal to the first component of x_{t-2k+1}
3. The fourth component *stochastically* copies the second part (consisting of $2k$ random bits) from one step earlier. Formally, each component $z_{t+1}^{(i)}$ is determined as follows: First take a sample $u_{t+1}^{(i)}$ from $Bernoulli(\frac{1}{4k})$, independent from all preceding observations. If $u_{t+1}^{(i)} = 1$, set $z_{t+1}^{(i)}$ to be equal to the second component of $w_t^{(i)}$. Otherwise, let $z_{t+1}^{(i)}$ be a fresh draw from $Bernoulli(0.5)$.

Predicting observations optimally requires taking into account observations from the $2k$ last time steps.

We show that, when approximately predicting with low memory capacities, a window-based approach does *not* in general achieve an optimal memory-surprisal tradeoff.

Consider a model that predicts x_{t+1} from only the last observation x_t , i.e., uses a window of length one. The only relevant piece of information in this past observation is w_t , which stochastically influences z_{t+1} . Storing this costs $2k$ bit of memory as w_t consists of $2k$ draws from $Bernoulli(0.5)$. How much does it reduce the surprisal of x_{t+1} ? Due to the stochastic nature of z_{t+1} , it reduces the surprisal only by about $I[x_{t+1}, w_t] = I[z_{t+1}, w_t] < 2k \cdot \frac{1}{2k} = 1$, i.e., surprisal reduction is strictly less than one bit.¹

We show that there is an alternative model that strictly improves on this window-based model: Consider a memory encoding model that encodes each of v_{t-2k+1}, \dots, v_t , which costs $2k$ bits of memory – as the window-based model did. Since $y_{t+1} = v_{t-2k+1}$, this model achieves a surprisal reduction of $H[v_{t-2k+1}] = 1$ bit, strictly more than the window-based model.

This result does not contradict our theorem because the theorem only provides *bounds* across models, which are not necessarily achieved by a given window-based model. In fact, for the process described here, no memory encoding function M can exactly achieve the theoretical bound described by the theorem.

2.2 Tight Bound for Retrieval Model

Here, we provide an example where our bound is tight for the retrieval-based model (Section 1.3) even though it is quite loose for the capacity model. That means, while no memory encoding function can exactly achieve the bound in the *capacity-bounded* setting for this particular stochastic process, there are *retrieval-based* memory encoding functions that exactly achieve the bound in the retrieval-based setting.

¹We can evaluate $I[z_{t+1}, w_t]$ as follows. Set $l = k/4$. Write z, w for any of the $2k$ components of z_{t+1}, w_t , respectively. First, calculate $p(z = 1|w = 1) = 1/l + (1 - 1/l)\frac{1}{2} = 1/(2l) + 1/2 = \frac{1+l}{2l}$ and $p(z = 0|w = 1) = (1 - 1/l)\frac{1}{2} = 1/2 - 1/2l = \frac{l-1}{2l}$. Then $I[Z, W] = D_{KL}(p(z|w = 1)||p(z)) = \frac{1+l}{2l} \log \frac{1+l}{1/2} + \frac{l-1}{2l} \log \frac{l-1}{1/2} = \frac{1+l}{2l} \log \frac{1+l}{1} + \frac{l-1}{2l} \log \frac{l-1}{1} \leq \frac{1+l}{1} \log \frac{1+l}{1} = (1 + 1/l) \log(1 + 1/l) \leq (1 + 1/l)(1/l) = 1/l + 1/l^2 < 2/l = \frac{1}{2k}$.

Defining the Process Let k be a positive integer. Consider a process $x_{t+1} = (y_{t+1}, z_{t+1}, u_{t+1}, v_{t+1})$ where

1. y_{t+1} consists of $2k$ random bits.
2. z_{t+1} is a draw from $Bernoulli(\frac{1}{4k^2})$.
3. u_{t+1} consists of $2k$ random bits if $z_t = 0$ and is equal to y_{t-2k+1} else.
4. $v_{t+1} := z_t$

Informally, z_t indicates whether u_{t+1} is copied from the past or a fresh sample; large values of k correspond to the setting where copying from the past only happens rarely.

Capacity Model We analyze the memory-surprisal tradeoff in the situation where prediction is optimal. Predicting observations x_{t+1}, x_{t+2}, \dots optimally from the past requires storing y_{t-2k+1}, \dots, y_t and z_t . This amounts to

$$H_M = (2k+1) \cdot 2k + H_2[1/4k^2] \geq 4k^2 \quad (59)$$

bits of memory in the capacity-based model, where $H_2[p] := -(p \log p + (1-p) \log(1-p))$.

We now evaluate I_t . We have

$$I_1 = I[v_{t+1}, z_t] = H_2[1/4k^2] \quad (60)$$

$$I_{2k} = I[x_{t+1}, x_{t-2k+1} | x_{t-2k+2} \dots x_t] = I[u_{t+1}, y_{t-2k+1} | z_{t+1}] = \frac{1}{4k^2} I[u_{t+1}, y_{t-2k+1} | z_{t+1} = 1] = \frac{2k}{4k^2} = \frac{1}{2k} \quad (61)$$

and all other values of I_t are zero.

Therefore, the theorem bounds the memory cost, in the limit of perfect prediction ($T \rightarrow \infty$), only by

$$H_M \geq \sum_{t=1}^{\infty} t I_t = 2k I_{2k} = 1 \quad (62)$$

compared to a true cost $H_M \geq 4k^2$. The bound provided by the theorem is therefore loose in this case for the capacity-based model.

Retrieval Model However, it is tight for the retrieval-based model. Again, we show this in the setting of optimally precise prediction. We use

$$s_t := (y_{t-2k+1}, \dots, y_t) \quad (63)$$

$$m_{t+1} := z_t \quad (64)$$

Then, if $z_t = 1$, we retrieve

$$r_t = g(x_{t-1}, m_{t-1}, s_{t-1}) := y_{t-2k+1} \quad (65)$$

Otherwise, if $z_t = 0$, we retrieve nothing. The cost of storing z_t is $H_2[1/4k^2]$, and the cost of retrieving r_t is $\frac{1}{4k^2} \cdot 2k = \frac{1}{2k}$.

In total, $H[m_t] = H_2[1/4k^2]$ and $H[r_t] = 1/2k$.

Taking, in the theorem, $T = 1$ and $S \rightarrow \infty$, we obtain

$$H[m_t] \geq I_1 = H_2[1/4k^2] \quad (66)$$

$$H[r_t] \geq I_{2k} = 1/2k \quad (67)$$

Thus, the bound is tight for both working memory and retrieval costs.

Furthermore, the bound provided by the theorem for the capacity-based model, while it can be loose for specific processes, is the tightest possible bound that only depends on the values of I_t . As the retrieval-based model is a generalization of the capacity-based model, it may be possible for the retrieval-based model to achieve the bound provided by the theorem even in cases when it is not possible for the capacity-based model.

2.3 Low memory requirements do not imply decay of unconditional mutual information

Our theoretical results link the memory-surprisal tradeoff to the values of *conditional* mutual information I_t , whereas prior work on the statistics of language has considered *unconditional* mutual information $I[w_t, w_0]$. Here, we show that the decay of unconditional mutual information is not necessarily linked to memory demands.

First, there are processes where unconditional mutual information does not decay with distance, even though memory load is small. Consider the constant process where with probability 1/2 all $w_t = 0$, and with probability 1/2 all $w_t = 1$. The unconditional mutual information is $I[w_t, w_0] = 1$ at all distances t , so does not decay at all. However, predicting the process optimally only requires 1 bit of memory. This is correctly captured by the Information Locality Theorem, as $I_1 = 1$ and $I_t = 0$ for $t > 1$, so $\lim_{T \rightarrow \infty} \sum_{t=1}^T t I_t = 1$.

Second, one can construct processes where the unconditional mutual informations $I[w_t, w_0]$ are zero for all distances t , but where optimal prediction requires nonzero memory: Consider the process consisting of 2 random bits and their XOR (called RRXOR by Crutchfield and Feldman, 2003). This one has nonzero I_2 , but zero unconditional mutual information $I[w_t, w_0]$ at all distances t . Conditional mutual information is not zero, however, and – in accordance with the Information Locality Theorem – optimal prediction requires at least $\lim_{T \rightarrow \infty} \sum_{t=1}^T t I_t > 0$ bits of memory (Crutchfield and Feldman, 2003).

3 Study 2

3.1 Corpus Size per Language

Language	Training	Held-Out	Language	Training	Held-Out
Afrikaans	1,315	194	Indonesian	4,477	559
Amharic	974	100	Italian	17,427	1,070
Arabic	21,864	2,895	Japanese	7,164	511
Armenian	514	50	Kazakh	947	100
Bambara	926	100	Korean	27,410	3,016
Basque	5,396	1,798	Kurmanji	634	100
Breton	788	100	Latvian	4,124	989
Bulgarian	8,907	1,115	Maltese	1,123	433
Buryat	808	100	Naija	848	100
Cantonese	550	100	North Sami	2,257	865
Catalan	13,123	1,709	Norwegian	29,870	4,639
Chinese	3,997	500	Persian	4,798	599
Croatian	7,689	600	Polish	6,100	1,027
Czech	102,993	11,311	Portuguese	17,995	1,770
Danish	4,383	564	Romanian	8,664	752

Dutch	18,310	1,518	Russian	52,664	7,163
English	17,062	3,070	Serbian	2,935	465
Erzya	1,450	100	Slovak	8,483	1,060
Estonian	6,959	855	Slovenian	7,532	1,817
Faroese	1,108	100	Spanish	28,492	3,054
Finnish	27,198	3,239	Swedish	7,041	1,416
French	32,347	3,232	Thai	900	100
German	13,814	799	Turkish	3,685	975
Greek	1,662	403	Ukrainian	4,506	577
Hebrew	5,241	484	Urdu	4,043	552
Hindi	13,304	1,659	Uyghur	1,656	900
Hungarian	910	441	Vietnamese	1,400	800

Table 2: Languages, with the number of training and held-out sentences available.

3.2 Details for Neural Network Models

The network is parameterized by a vector θ of weights determining how the activations of neurons propagate through the network (Hochreiter and Schmidhuber, 1997). Given a corpus, the numeral parameters of the LSTM are chosen so as to minimize the average surprisal across the training corpus. At the beginning of training, the parameters θ are randomly initialized to some setting θ_0 .

The training corpus is chopped into word sequences $w_1 \dots w_{T_{max}}$ of length T_{max} , where T_{max} is the highest T for which we estimate I_T . We use Stochastic Gradient Descent to optimize the parameters θ so as to minimize the surprisal

$$\frac{1}{T_{max}} \sum_{i=1}^{T_{max}} \log p_{\theta}(w_i | w_1 \dots w_{i-1}) \quad (68)$$

When calculating the parameter update, we use three standard methods of regularization that have been shown to improve neural language modeling: dropout (Srivastava et al., 2014), word dropout, and word noising (Xie et al., 2017).

Once all sequences have been processed, we start another pass through the training data. Before each pass through the training data, the order of sentences of the training data is shuffled, and the corpus is again chopped into sequences of length T . After each pass through the training data, the average surprisal (68) at the current parameter setting θ is evaluated on the held-out partition. We terminate training once this held-out surprisal does not improve over the one computed after the previous pass any more.

In our experiments, we chose $T_{max} = 20$. Prior work has found that the probabilities $p(w_t | w_1 \dots w_{t-1})$ are dominated by a small number of preceding words (Daniluk et al., 2017), suggesting that I_t will be close to zero for t greater than 20.

3.2.1 Choice of Hyperparameters

The LSTM model has a set of numerical *hyperparameters* that need to be specified before parameter estimation, such as the number of neurons and the learning rate. For each corpus, we used Bayesian optimization using the Expected Improvement acquisition function (Snoek et al., 2012) to find a good setting of the hyperparameters. We optimized the hyperparameters to minimize average surprisal (68) on the held-out partition

resulting at the end of parameter estimation, on languages generated from random word order grammars. This biases the hyperparameters towards modeling counterfactual grammars better, biasing them *against* our hypothesis that real orders result in better memory-surprisal tradeoffs than counterfactual orders.

Due to reasons of computational efficiency, neural language models can only process a bounded number of distinct words in a single language (Mikolov et al., 2010). For each corpus, we limited the number of distinct processed words to the $N = 10,000$ most common words in the training corpus, a common choice for neural language models. We represented other words by their part-of-speech tags as annotated in the corpora. This applied to 37 languages, affecting an average of 11 % of words in these languages. We believe that this modeling limitation does not affect our results for the following reasons. First, this affects the same words in real and counterfactually ordered sentences. Second, all excluded words are extremely infrequent in the available data, occurring less than 10 times (except for Czech and Russian, the languages for which we have by far the largest datasets). Many of the excluded words occur only once in the dataset (78 % on average across the affected languages). This means that any model would only be able to extract very limited information about these words from the available training data, likely *less* than what is provided by the part-of-speech tag. Third, traditional N-gram models, which do not have this limitation, provide results in qualitative agreement with the neural network-based estimates.

3.2.2 Estimation of average surprisal

As described in the main paper, the mutual information I_t is estimated from entropies obtained with Markov models:

$$S_t = H[w_t | w_0, \dots, w_{t-1}]$$

We estimate these entropies as follows. After estimating the parameter vector θ , we compute the following (T ranging from T_{max} up to the length of the held-out partition) in the held-out partition:

$$\widehat{S}_T = \frac{1}{|HeldOut| - T} \sum_{i=T}^{|HeldOut|} \log P_{\theta}[w_i | w_{i-T}, w_{i-T+1}, \dots, w_{i-1}] \quad (69)$$

where $|HeldOut|$ is the number of words in the held-out set.

For larger values of T , the model may overfit, leading to estimates where \widehat{S}_T may *increase* as the context size increases. Such a situation is an artifact of overfitting, and cannot happen for the true entropies S_t . Directly estimating I_t from \widehat{S}_T would lead to negative estimates of I_t , again impossible for the true values of this quantity. We eliminate this pathological behavior by only estimating

$$S_t \approx \min_{s \leq t} \widehat{S}_s, \quad (70)$$

which amounts to only considering higher-order models $P_{\theta}[w_t | w_{t-T}, w_{t-T+1}, \dots, w_{t-1}]$ when they improve over lower-order ones. This procedure ensures that \widehat{S}_t can only decrease as the context size t increases.

For each language, we collected data from the actual orderings and from several random grammars. We collect multiple samples for the actual orderings to control for variation due to the random initialization of the neural network. For each of the random grammars, we collect one sample. Data is collected according to a precision-based stopping criterion described in Section 3.2.3.

We estimate the unigram entropy $H[w_0]$ by averaging over all model runs on a given corpus.

3.2.3 Number of Samples, Precision-Based Stopping Criterion

Training neural language models is computationally costly. Therefore, we used a precision-based stopping criterion to adaptively choose a sample size for each language. Precision-based stopping criteria offer a way to adaptively choose sample size without biasing results for or against the hypothesis of interest.

We propose a stopping criterion using a global measure of the degree of optimization of the real language. For each sample x from real orderings, we look at the proportions $N_+(x)$ of samples from the baseline languages that are more optimal than x throughout the entire range where both curves are defined, and the proportion $N_-(x)$ of baseline samples that are consistently less optimal. We estimate the quotient

$$G := \frac{\mathbb{E}_{x \sim P_1} [N_+(x)]}{\mathbb{E}_{x \sim P_1} [N_+(x) + N_-(x)]} \quad (71)$$

where P_1 is the distribution over values obtained for real orderings. We use a bootstrapped confidence interval for $\mathbb{E}[G]$ for quantifying the degree of optimization. For bootstrapping, we separately resample samples from the real language and from the baseline grammars. Due to the use of bootstrapping, the confidence intervals are not exact.

For each language, we first collected 10 data points for real orderings and 10 data points for baseline orderings. We continued obtaining new data points until the CI for G had width ≤ 0.15 , or there were 100 samples from P_1 and 300 samples from P_2 . Up to the end, we chose the next sample to be from P_0 with probability $2/3$, and P_1 otherwise.²

This procedure was parallelized on several machines. In the case where the stopping criterion was reached for a language while several machines were still computing samples for this language, we did not discard those samples. Consequently, more samples were collected than necessary to reach the stopping criterion; however, in a way that does not bias our results towards or against our hypothesis.

3.3 Samples Drawn per Language

Language	Base.	Real	Language	Base.	Real
Afrikaans	13	10	Indonesian	11	11
Amharic	137	10	Italian	10	10
Arabic	11	10	Japanese	25	15
Armenian	140	76	Kazakh	11	10
Bambara	25	29	Korean	11	10
Basque	15	10	Kurmanji	338	61
Breton	35	14	Latvian	308	178
Bulgarian	14	10	Maltese	30	24
Buryat	26	18	Naija	214	10
Cantonese	306	32	North Sami	335	194
Catalan	11	10	Norwegian	12	10
Chinese	21	10	Persian	25	12
Croatian	30	17	Polish	309	35
Czech	18	10	Portuguese	15	55
Danish	33	17	Romanian	10	10
Dutch	27	10	Russian	20	10

²Due to a scripting error, a much higher number of samples was generated for Erzya.

English	13	11	Serbian	26	11
Erzya	846	167	Slovak	303	27
Estonian	347	101	Slovenian	297	80
Faroese	27	13	Spanish	14	10
Finnish	83	16	Swedish	31	14
French	14	11	Thai	45	19
German	19	13	Turkish	13	10
Greek	16	10	Ukrainian	28	18
Hebrew	11	10	Urdu	17	10
Hindi	11	10	Uyghur	326	175
Hungarian	220	109	Vietnamese	303	12

Figure 1: Samples drawn per language according to the precision-dependent stopping criterion.

Language	Mean	Lower	Upper	Language	Mean	Lower	Upper
Afrikaans	1.0	1.0	1.0	Indonesian	1.0	1.0	1.0
Amharic	1.0	1.0	1.0	Italian	1.0	1.0	1.0
Arabic	1.0	1.0	1.0	Japanese	1.0	1.0	1.0
Armenian	0.92	0.87	0.97	Kazakh	1.0	1.0	1.0
Bambara	1.0	1.0	1.0	Korean	1.0	1.0	1.0
Basque	1.0	1.0	1.0	Kurmanji	0.93	0.88	0.98
Breton	1.0	1.0	1.0	Latvian	0.49	0.4	0.57
Bulgarian	1.0	1.0	1.0	Maltese	1.0	1.0	1.0
Buryat	1.0	1.0	1.0	Naija	1.0	0.99	1.0
Cantonese	0.96	0.86	1.0	North Sami	0.37	0.3	0.44
Catalan	1.0	1.0	1.0	Norwegian	1.0	1.0	1.0
Chinese	1.0	1.0	1.0	Persian	1.0	1.0	1.0
Croatian	1.0	1.0	1.0	Polish	0.1	0.04	0.17
Czech	1.0	1.0	1.0	Portuguese	1.0	1.0	1.0
Danish	1.0	1.0	1.0	Romanian	1.0	1.0	1.0
Dutch	1.0	1.0	1.0	Russian	1.0	1.0	1.0
English	1.0	1.0	1.0	Serbian	1.0	1.0	1.0
Erzya	0.99	0.98	1.0	Slovak	0.07	0.03	0.12
Estonian	0.8	0.72	0.86	Slovenian	0.82	0.77	0.88
Faroese	1.0	1.0	1.0	Spanish	1.0	1.0	1.0
Finnish	1.0	1.0	1.0	Swedish	1.0	1.0	1.0
French	1.0	1.0	1.0	Thai	1.0	1.0	1.0
German	1.0	0.91	1.0	Turkish	1.0	1.0	1.0
Greek	1.0	1.0	1.0	Ukrainian	1.0	1.0	1.0
Hebrew	1.0	1.0	1.0	Urdu	1.0	1.0	1.0
Hindi	1.0	1.0	1.0	Uyghur	0.65	0.57	0.73
Hungarian	0.87	0.8	0.93	Vietnamese	1.0	0.98	1.0

Figure 2: Bootstrapped estimates for the precision-dependent stopping criterion G .

3.4 N-Gram Models

Here we show that the results of Study 2 remain robust when estimating surprisal with a simple n -gram model instead of recurrent neural networks.

3.4.1 Method

We use a version of Kneser-Ney Smoothing (Kneser and Ney, 1995). For a sequence $w_1 \dots w_k$, let $N(w_{1\dots k})$ be the number of times $w_{1\dots k}$ occurs in the training set. The unigram probabilities are estimated as

$$p_1(w_t) := \frac{N(w_t) + \delta}{|Train| + |V| \cdot \delta} \quad (72)$$

where $\delta \in \mathbb{R}_+$ is a hyperparameter. Here $|Train|$ is the number of tokens in the training set, $|V|$ is the number of types occurring in train or held-out data. Higher-order probabilities $p_t(w_t|w_{0\dots t-1})$ are estimated recursively as follows. Let $\gamma > 0$ be a hyperparameter. If $N(w_{0\dots t-1}) < \gamma$, set

$$p_t(w_t|w_{0\dots t-1}) := p_{t-1}(w_t|w_{1\dots t-1}) \quad (73)$$

Otherwise, we interpolate between t -th order and lower-order estimates:

$$p_t(w_t|w_{0\dots t-1}) := \frac{\max(N(w_{0\dots t}) - \alpha, 0.0) + \alpha \cdot \#\{w : N(w_{0\dots t-1}w) > 0\} \cdot p_{t-1}(w_t|w_{1\dots t-1})}{N(w_{0\dots t-1})} \quad (74)$$

where $\alpha \in [0, 1]$ is also a hyperparameter. Kneser and Ney (1995) show that this definition results in a well-defined probability distribution, i.e., $\sum_{w \in V} p_t(w|w_{0\dots t-1}) = 1$.

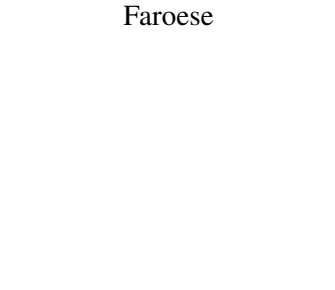
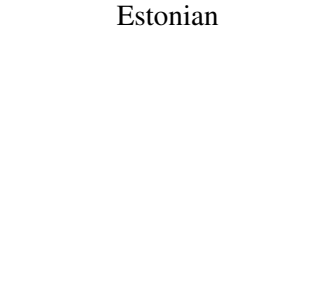
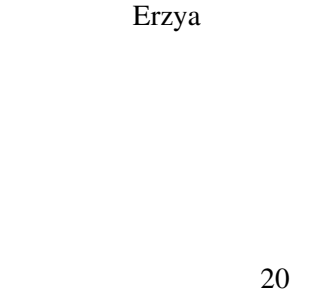
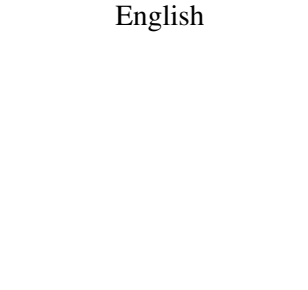
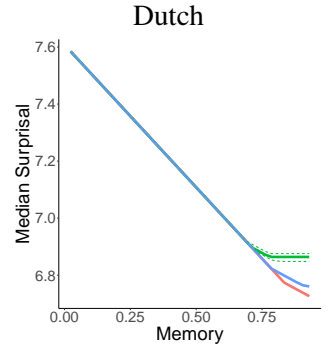
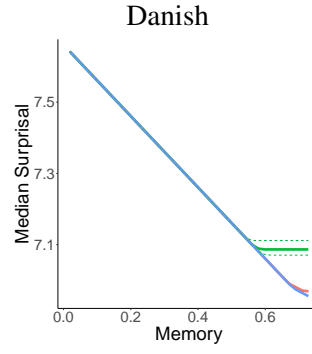
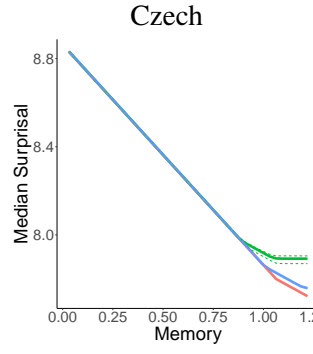
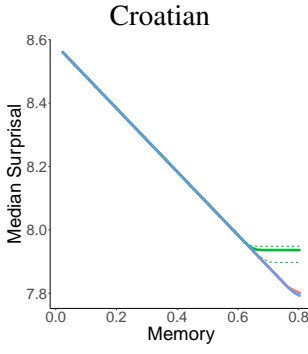
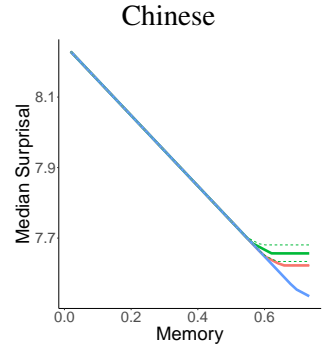
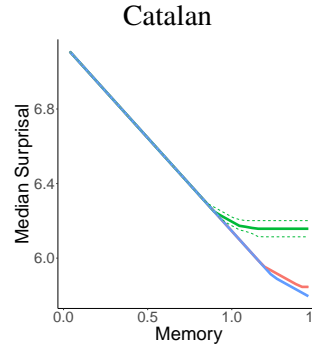
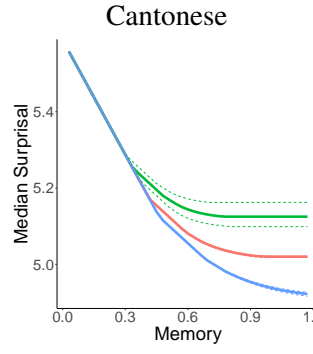
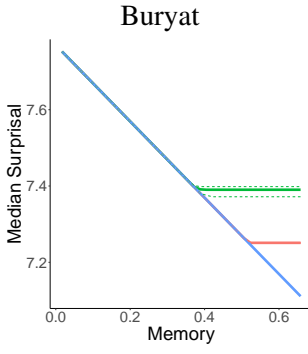
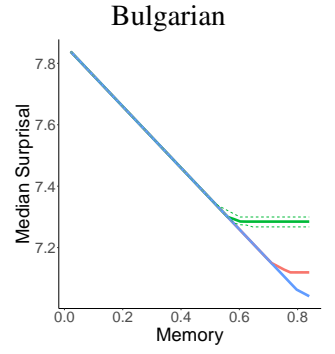
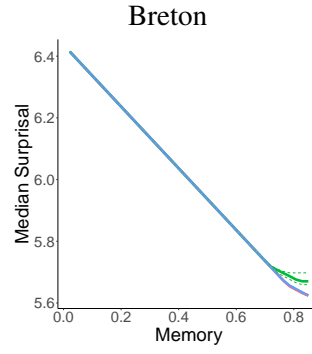
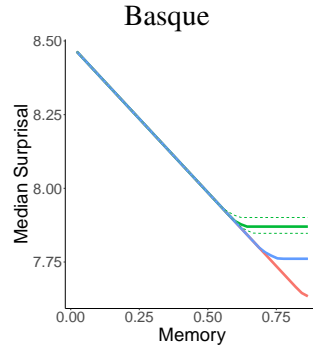
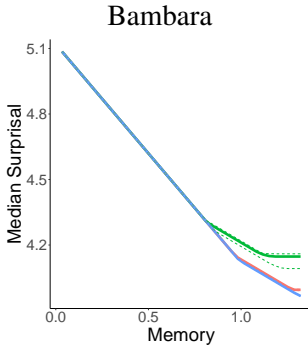
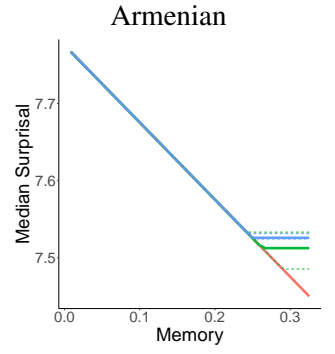
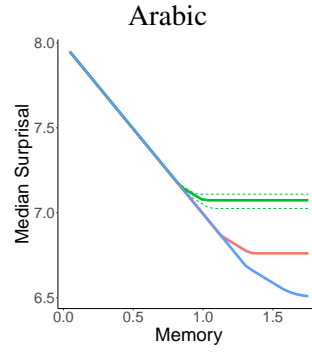
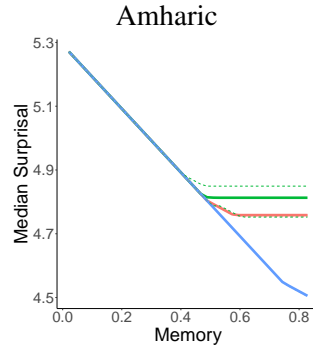
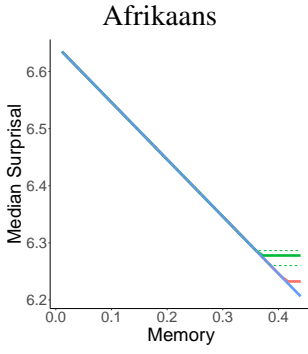
Hyperparameters α, γ, δ are tuned using the held-out set, with the same strategy as for the neural network models.

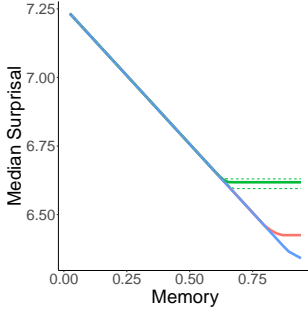
3.4.2 Results

Resulting tradeoff curves are shown in Figure 3, for real orders (blue), random baselines (red), and ordering grammars fitted to the observed orders (green).

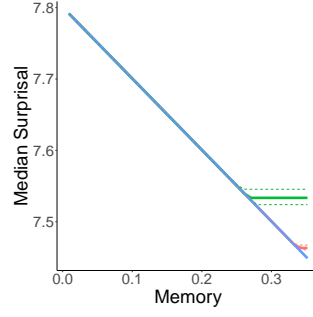
In five languages (Polish, Slovak, North Sami, Armenian, Latvian), AUC is numerically higher for the real orders than for at least 50% of baseline grammars. Among the remaining 49 languages, AUC is significantly lower than for at least 50% of baseline grammars in 46 languages at $p = 0.01$, where we controlled for multiple comparisons using Hochberg’s step-up procedure. In three languages (German, Faroese, Kurmanji), the difference is numerical but not significant in this analysis. In 44 languages, the real order has lower AUC than 100% of sampled baseline grammars.

The main divergence in these results from those of the neural network-based estimator in the main paper is that a few languages with small corpora (Armenian, Faroese, Kurmanji) and a language with flexible word order (German) do not show clear evidence for optimization for the simple n -gram estimator. In the other languages, results qualitatively agree with those of the neural network-based estimator.

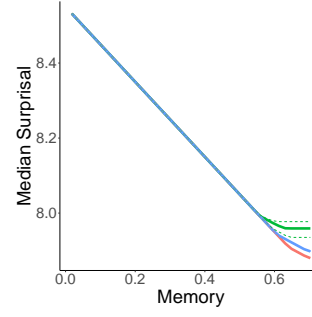




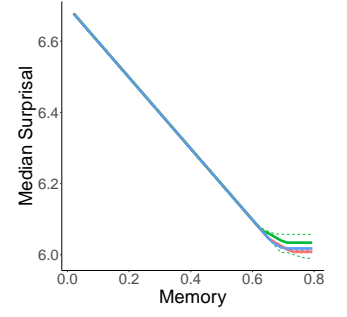
Finnish



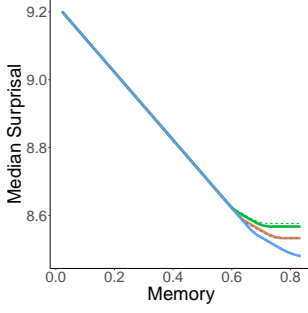
French



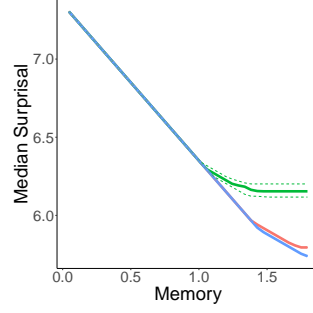
German



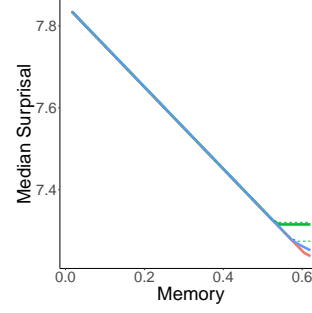
Greek



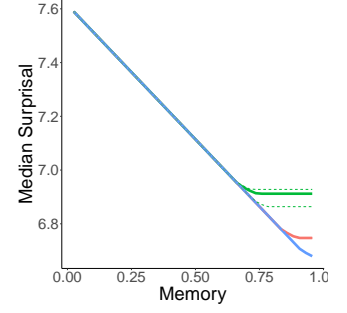
Hebrew



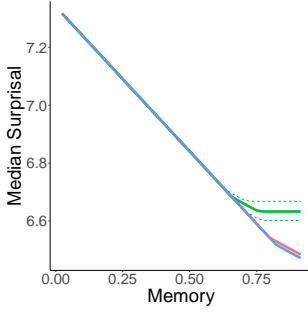
Hindi



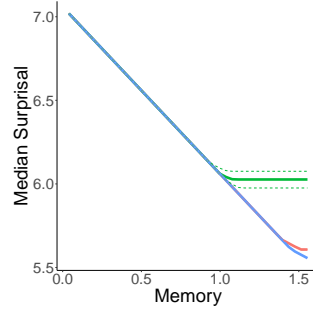
Hungarian



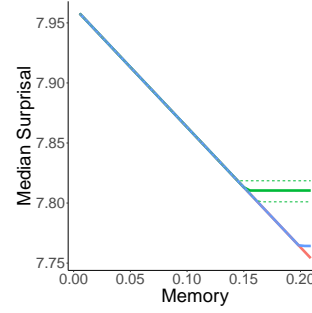
Indonesian



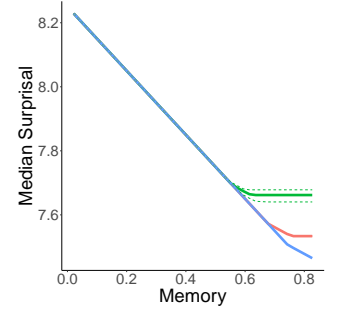
Italian



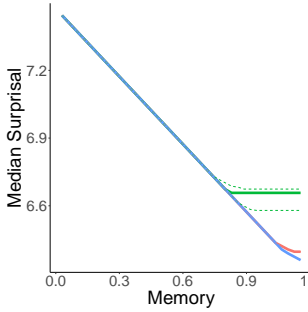
Japanese



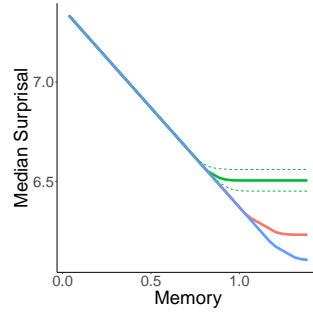
Kazakh



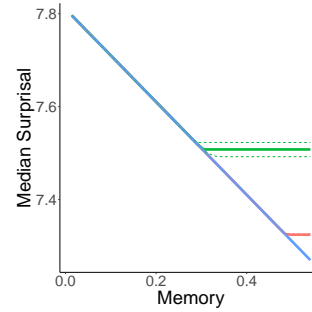
Korean



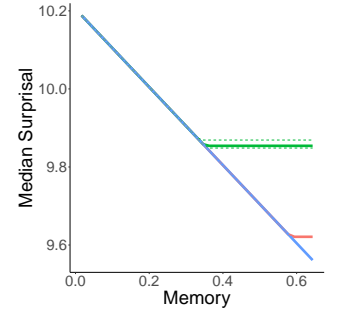
Kurmanji



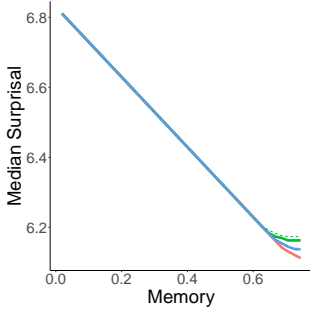
Latvian



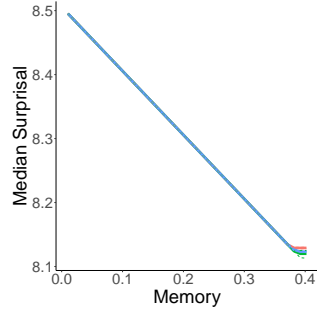
Maltese



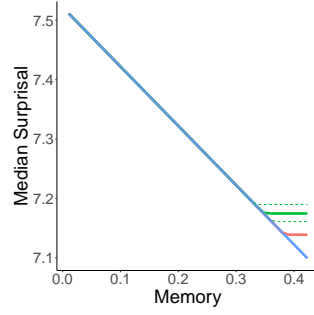
Naija



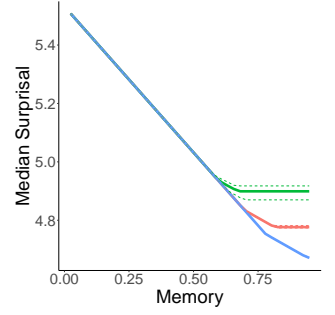
North Sami



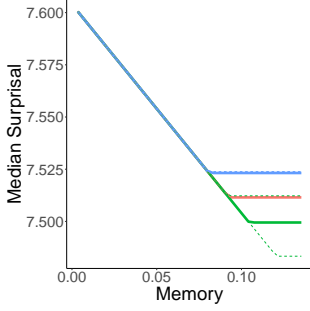
Norwegian



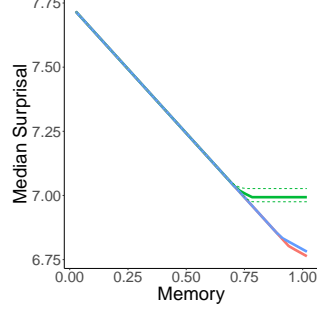
Persian



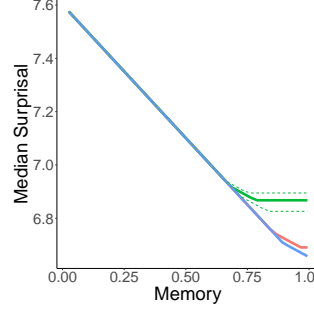
Polish



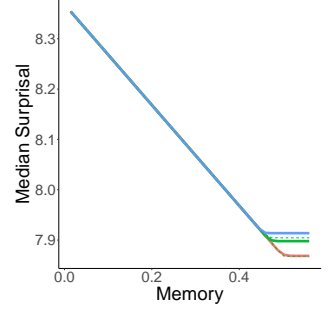
Portuguese



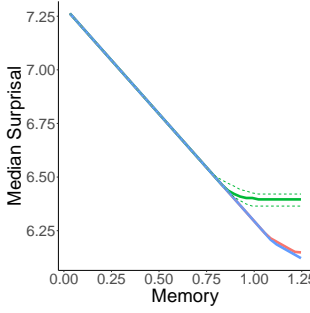
Romanian



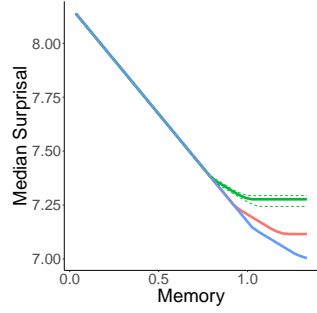
Russian



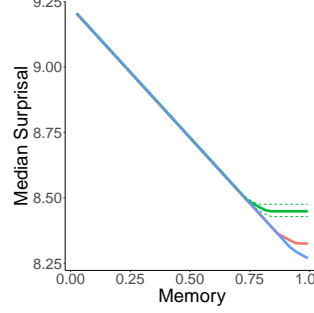
Serbian



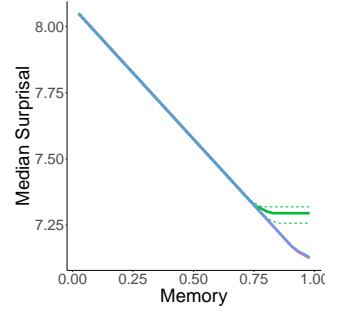
Slovak



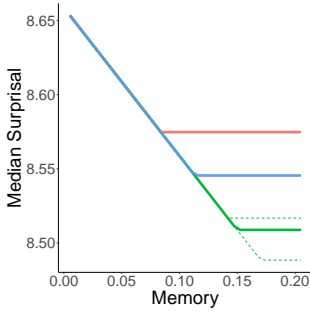
Slovenian



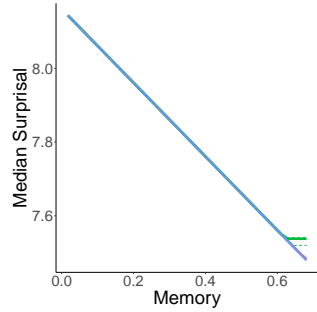
Spanish



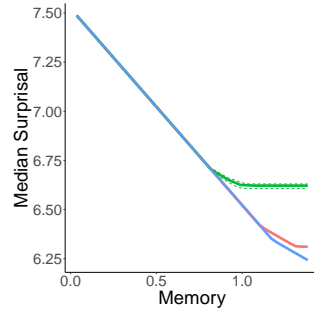
Swedish



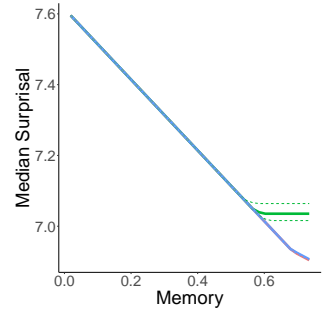
Thai



Turkish



Ukrainian



Urdu

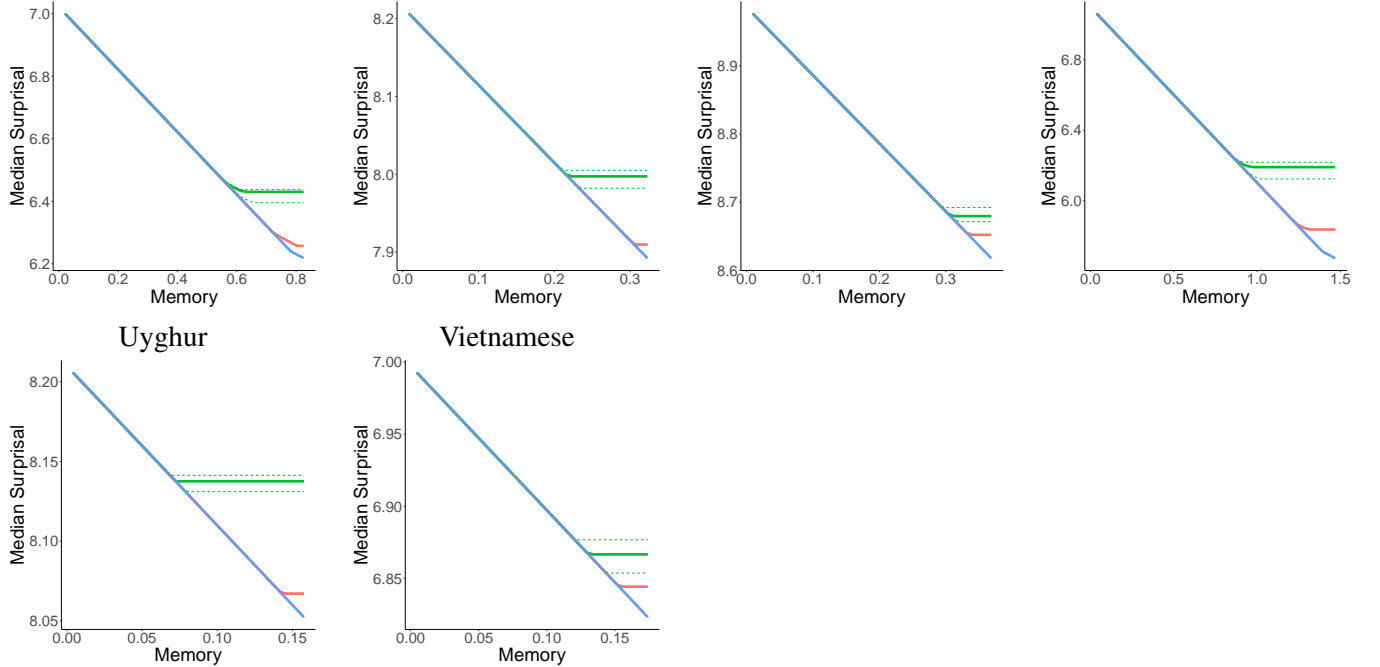


Figure 3: Memory-surprisal tradeoff curves (estimated using n -gram models): For each memory budget, we provide the median surprisal for real and random languages. Solid lines indicate sample medians for n -grams, dashed lines indicate 95 % confidence intervals for the population median. Red: Random baselines; blue: real language; green: maximum-likelihood grammars fit to real orderings.

3.5 Chart Parsing Control

LSTMs and n -gram models are linear sequence models that might incorporate biases towards linear order as opposed to hierarchical structure. In particular, this might bias these models towards modeling relations between elements better when they are close in linear order. Here we use chart parsing to show that the results also hold when estimating I_t using a model that is based on hierarchical structure and incorporates no bias towards linear closeness.

We use probabilistic context-free grammars (PCFG), a common formalism for representing probability distributions based on syntactic structure. PCFG surprisal is often computed in psycholinguistic research using approximate incremental parsers (Roark, 2001; Demberg et al., 2013; Schijndel et al., 2013), but these might themselves incorporate some biases towards linear closeness due to the use of techniques such as beam-search and pruning. We instead opt for exact inference for PCFGs using chart parsing, which computes exact probabilities and surprisals for a given PCFG.

3.5.1 Deriving PCFGs from Dependency Corpora

Here, we describe how we constructed a PCFG from the training section of a dependency corpus. There is no universally accepted standard method of extracting PCFGs from dependency corpora; we chose the following procedure that tries to balance between preserving information about dependency structure and keeping the size of grammars computationally manageable.

In a first step we convert the dependency trees into binary constituent trees. We binarize so that left children branch off before right children. We assign nonterminal labels to the resulting constituents as follows. Preterminals are labeled with (1) the POS of the head, and (2) its lexical identity. We assign nonterminal labels to constituents spanning more than one word based on (1) the POS of the head, (2) the lexical identity of the head, (3) the dependency label linking head and dependent. These choices are driven by the desire to preserve information about the dependency structure in the constituent trees.

In a second step, it is necessary to reduce the number of preterminals and nonterminals, both to deal with data sparsity, and to make chart parsing tractable. In our implementation for calculating I_t (see below), we found that up to 700 nonterminals were compatible with efficient inference. (For comparison, the Berkeley parser as described by Petrov and Klein (2007) uses 1,090 nonterminals for its English grammar, while employing a highly optimized coarse-to-fine strategy that includes pruning, and thus does not provide exact inference for surprisal estimation.) We reduced the number of nonterminals as follows: (1) For words with frequency below a threshold parameter, we did not record lexical identity in preterminals and nonterminals. (2) Nonterminals that only differ in the relation label were merged if their frequency fell below a threshold parameter, (2) Nonterminals that only differ in the head’s lexical identity were merged if their frequency fell below a threshold parameter. Furthermore, words occurring less than 3 times in the dataset were replaced by OOV.

An alternative method to reduce the number of nonterminals is to use merge-and-split (Petrov and Klein, 2007), but that method would have taken too long to run on all the 54 corpora.

We chose the threshold parameters for (1)-(3) separately for each language by sampling 15 configurations, and choosing the one that minimized estimated surprisal (see below) on a sampled baseline grammar, while resulting in at most 700 nonterminals and preterminals.

An alternative estimation method avoiding the binarization step would be to use the Earley parser, but that would have made it difficult to parallelize processing on GPUs (see below).

3.5.2 Estimating I_t with Chart Parsing

Calculating I_t requires estimating entropies $H[w_1, \dots, w_t]$, and thus probabilities $P(w_1, \dots, w_t)$. This is challenging because it requires marginalization over possible positions in a sequence. The standard parsing algorithm for binary PCFGs is the CKY algorithm; however, the standard form of this algorithm only computes the surprisal for entire sentences. There is a known extension of the CKY algorithm that calculates *prefix* probabilities (Jelinek and Lafferty, 1991; Stolcke, 1995; Goodman, 1999):

$$P[\#, X_1, \dots, X_t] := \sum_N \sum_{Y_{1..N}} P(\#, X_1, \dots, X_t, Y_{1..N}, \#) \quad (75)$$

(here, # denotes the beginning/end of a sentence), that is, the probability mass assigned to all sentences starting with the given prefix X_1, \dots, X_t .

However, simultaneously summing over possible left *and* right continuations is more challenging.³ We approach this by restricting the summation on the left to prefixes of a fixed length:

$$\sum_{Y_1 \dots Y_N} P(\#, Y_1 \dots Y_N, X_1, \dots, X_t) \quad (76)$$

³Nederhof and Satta (2011) describe a method for calculating infix probabilities, but this method, besides being computationally costly due to construction of a large finite automaton, computes something subtly different from the quantity required here: It computes the probability mass of sentences containing a given string, not accounting for multiple occurrences of the same string in a longer sentence.

and estimating

$$P(X_t|X_1 \dots X_{t-1}) \approx \mathbb{E}_{Y_1 \dots Y_N} P(X_t|\#, Y_1 \dots Y_N, X_1, \dots, X_{t-1}) \quad (77)$$

Under certain conditions on the PCFG, this approximation provably converges to the true value for sufficiently large values of N . Empirically, we found that the values already became essentially stationary at $N \geq 5$.

For computational efficiency, we estimated I_t for $t = 1, \dots, 5$, finding I_t to be very close to zero for higher t . We ran the algorithm on all contiguous sequences of length $T = 5$. Following Kim et al. (2019), we took advantage of GPU parallelization for implementation of the CKY algorithm, processing 1,000 sequences in parallel.

3.5.3 Results

We computed I_t for the MLE grammar and for five random baseline grammars. We did not run this on the observed orderings, as these may have crossing branches, making binarization difficult and thus rendering comparison with baselines less meaningful.

The resulting memory-surprisal tradeoff bounds are shown in Figure 4. In most languages, a more efficient tradeoff curve is estimated for the fitted grammars than for the baseline grammars. In five languages (Finnish, Slovak, North Sami, Cantonese, Kurmanji), the fitted grammar numerically has higher AUC value than at least 50% of baseline grammars. In all other 49 languages the fitted grammar numerically has lower AUC than more than 50% of baseline grammars. In 37 languages, the fitted grammar has lower AUC than 100% of sampled baselines.

Note that absolute numbers are not comparable with other models because there are many out-of-vocabulary tokens (they are necessary because the number of non- and preterminals has to be kept low). Also, we note that the amount of exploited predictive information is much lower than in the other models, that is, the difference between surprisal at zero memory and surprisal at maximal memory is low. This agrees with the observation that PCFG independence assumptions are inadequate, and that chart parsers have not historically reached good perplexities (parsers with good perplexities such as Roark Parser and RNNs do not make these independence assumptions, but also do not allow efficient exact chart parsing). Nonetheless, the experiment confirms the finding with a model that is based on hierarchical syntactic structure while enabling exact inference.

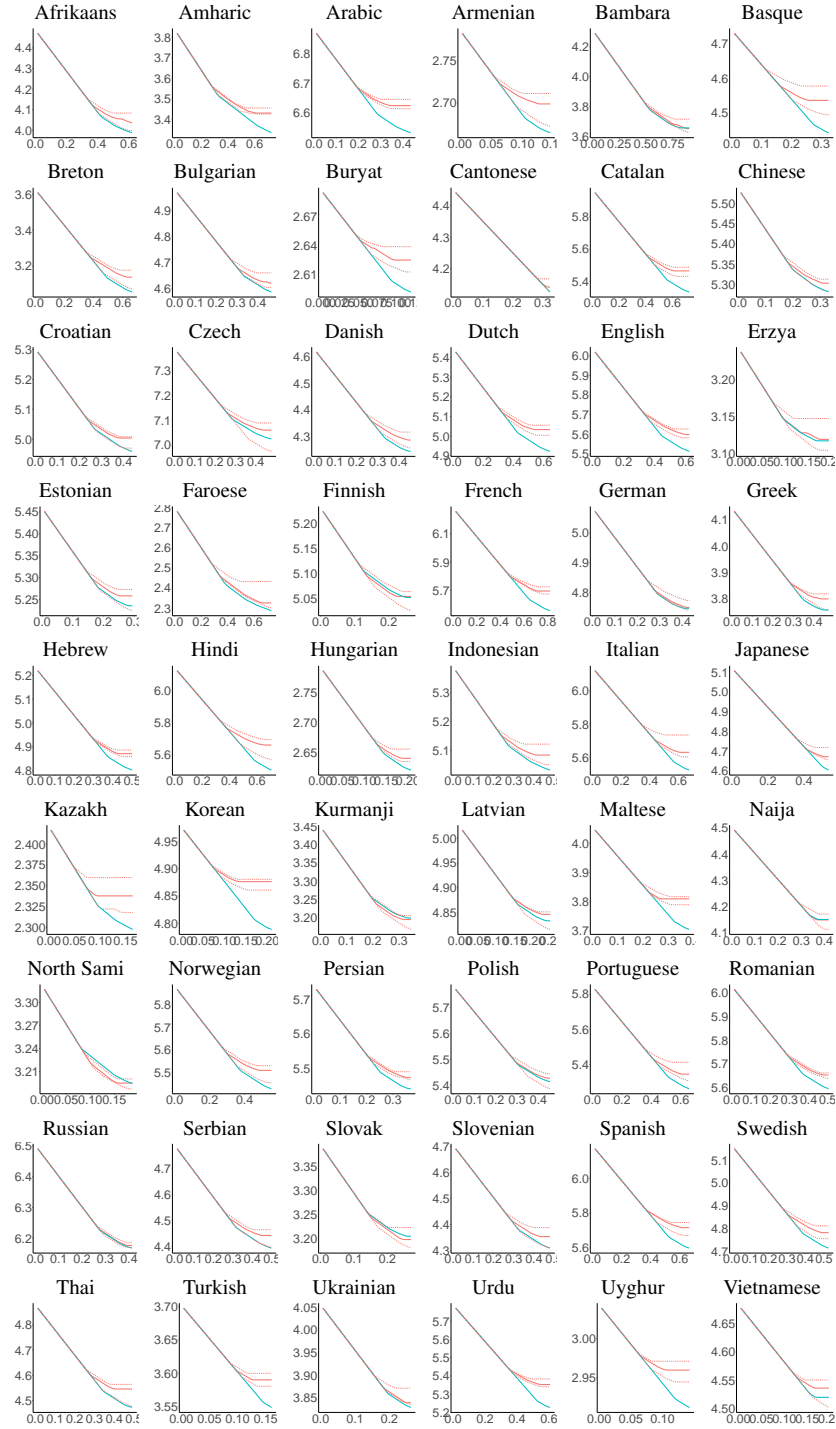


Figure 4: Memory-surprisal tradeoffs computed with the PCFG estimator, comparing fitted grammars (blue) with baselines (red). For the random baselines, we provide the sample median and 95% confidence intervals obtained with the binomial test.

4 Study 3

4.1 Determining Japanese Verb Suffixes

Here, we describe how we determined the Japanese verb suffixes described in the main paper. We determined a set of frequent morphemes as follows. We selected all morphemes occurring in the dataset at least 50 times and annotated their meaning/function. Among these, three morphemes are treated as independent words, not suffixes, by Kaiser et al. (2013) (*dekiru* ‘be able to’, *naru* ‘become’, *yoo* ‘as if’); we excluded these. Furthermore, passive and potential markers are formally identical for many verbs; we included both here.

We list the morphemes according to the order extracted according to the model. Note that there is no universally accepted segmentation for Japanese suffixes; we follow the UD tokenization in choosing which suffixes to segment.⁴

1. Derivation: *-su-* (allomorphs *-suru-*, *-shi-*), derives verbs from Sino-Japanese words. This is lemmatized as *suru*.
2. VALENCE: causative (*-(s)ase-*) (Hasegawa (2014, 142), Kaiser et al. (2013, Chapter 13)). In the UD data, this is lemmatized as *saseru*, *seru* (190 occurrences).
3. VOICE: passive (*-are-*, *-rare-*) (Hasegawa (2014, 152), Kaiser et al. (2013, Chapter 12)). In the UD data, this is lemmatized as *rareru*, *reru* (\approx 2000 occurrences).
4. MOOD, MODALITY:
 - (a) potential (allomorphs *-are-*, *-rare-*, *-e-*). In the UD data, this is lemmatized as *rareru*, *reru*, *eru*, *keru*. This is formally identical to the passive morpheme for many verbs (Vaccari and Vaccari (1938, 346), Kaiser et al. (2013, 398)).
 - (b) politeness *-mas-* (allomorphs *-masu-*, *-mashi-*, *-mase-*) (Kaiser et al., 2013, 190). In the UD data, this is lemmatized as *masu* (\approx 600 occurrences).
 - (c) MODALITY: desiderative *-ta-* (allomorphs: *-tai-*, *-taku-*, *-taka-*) (85 occurrences) (Kaiser et al., 2013, 238).
5. NEGATION: negation *-na-* (allomorphs: *-nai*, *-n-*, *-nakat-*). Lemmatized as *nai* (630 occurrences).
6. TENSE/ASPECT/MOOD:
 - (a) *-ta* for past (4K occurrences) (Kaiser et al., 2013, 211)
 - (b) *-yoo* for hortative, future, and similar meanings (Kaiser et al., 2013, 229). This is lemmatized as *u* (92 occurrences).
7. *-te* derives a nonfinite form (Kaiser et al., 2013, 186). (4K occurrences)

We provide examples illustrating the relative ordering of different morphemes. Note that passive and potential markers do not co-occur; we merge them here because they are not formally distinct for many verbs. We omit examples with *-te*; it always follows other suffixes that are compatible with it.

⁴The biggest difference to some other treatments is that the ending *-u/-ru* is viewed as part of the preceding morpheme that appears in some environments due to allomorphic variation, while it is viewed as a nonpast suffix in some other treatments (Hasegawa, 2014, p.116); if it were treated as a nonpast suffix, it would occupy a slot together with the past, future/hortative, and nonfiniteness affixes.

Stem	Caus.	Pass./Pot.	Polite.	Desid.	Neg.	TAM	
mi					naka	tta	did not see (Vaccari and Vaccari, 1938, 153)
mi				taku	nai		do not wish to see (Vaccari and Vaccari, 1938, 98)
mi				taku	naka	tta	did not wish to see (Vaccari and Vaccari, 1938, 98)
tat	ase	rare				ta	was made to stand up (Kaiser et al., 2013, 396)
waraw		are				ta	was laughed at (Kaiser et al., 2013, 384)
mi		rare	mase		n		is not seen (Vaccari and Vaccari, 1938, 337)
mi		rare	mash			yoo	will be seen (Vaccari and Vaccari, 1938, 337)
de					naka	roo	will not go out (Vaccari and Vaccari, 1938, 170)
mi		e	mase		n		cannot see (Vaccari and Vaccari, 1938, 349)

4.2 Determining Sesotho Verb Affixes

Here, we describe how we determined the Sesotho verb prefixes and suffixes. Sesotho has composite forms consisting of an inflected auxiliary followed by an inflected verb. Both verbs carry subject agreement. While they are annotated as a unit in the Demuth corpus, they are treated as separate words in grammars (Doke and Mofokeng, 1967; Guma, 1971). We separated these, taking the main verb to start at its subject agreement prefix. We only considered main verbs for the experiments here. Forms in child utterances are annotated with well-formed adult forms; we took these here. In the Demuth corpus, each morpheme is annotated; a one- or two-letter key indicates the type of morpheme (e.g. subject agreement, TAM marker). We classified morphemes by this annotation.

According to Demuth (1992), affixes in the Sesotho verb have the following order:

1. Subject agreement
2. Tense/aspect
3. Object agreement
4. Verb stem
5. ‘Extension’/perfect/passive markers, where ‘extension’ refers to causative, neuter/stative, reversive, etc.
6. Mood

We refined this description by considering all morpheme types occurring at least 50 times in the corpus.

As in Japanese, morphemes show different forms depending on their environment. The corpus contains some instances of fused neighboring morphemes that were not segmented further; we segmented these into their underlying morphemes for modeling prediction on the level of morphemes.

Prefixes

1. Subject agreement:

This morpheme encodes agreement with the subject, for person, number, and noun class (the latter only in the 3rd person) (Doke and Mofokeng, 1967, §395) (Guma, 1971, p. 162).

In the Demuth corpus, this is annotated as *sm* (17K occurrences) for ordinary forms, and *sr* (193 occurrences) for forms used in relative clauses.

2. Negation:

In various TAM forms, negation is encoded with a morpheme *-sa-* in this position (362 occurrences) (Guma, 1971, p. 172) (Doke and Mofokeng, 1967, §429). Common allomorphs in the corpus include *ska*, *seka*, *sa*, *skaba*.

3. Tense/Aspect/Mood, annotated as t^{\wedge} (13K occurrences) (Guma, 1971, p. 165)

Common TAM markers in this position in the corpus include, with the labels provided in the Demuth corpus:

- *-tla-*, *-tlo-*, *-ilo-* future (Doke and Mofokeng, 1967, §410–412)
- *-a-* present (Doke and Mofokeng, 1967, §400)
- *-ka-* potential (Doke and Mofokeng, 1967, §422–428)
- *-sa-* persistive (Doke and Mofokeng, 1967, §413–418)
- *-tswa-* recent past (Doke and Mofokeng, 1967, §404–406)

In the corpus, TAM prefixes are often fused with the subsequent object marker.

4. OBJECT agreement (labeled *om*, 6K occurrences) or reflexive (labeled *rf*, 751 occurrences).

Similar to subject agreement, object agreement denotes person, number, and noun class features of the object. Unlike subject agreement, it is optional (Doke and Mofokeng, 1967, §459).

Object agreement and reflexive marking are mutually exclusive (Guma, 1971, p. 165).

Verb Suffixes in Sesotho Again, we extracted morpheme types occurring at least 50 times.

1. Reversive: (labeled *rv*, 214 occurrences), (Doke and Mofokeng, 1967, §345).

This suffix changes semantics. Examples: *tlama* ‘bind’ – *tlamōlla* ‘loosen’, *etsa* ‘do’ – *etsōlla* ‘undo’ (Doke and Mofokeng, 1967, §346). Such suffixes are found across Bantu languages (Schadeberg, 2003).

2. VALENCE:

- (a) causative (labeled *c*, 1K occurrences), *-isa* (with morphophonological changes) (Doke and Mofokeng, 1967, §325)
- (b) neuter (labeled *nt*, 229 occurrences), *-eha*, *-ahala* (Doke and Mofokeng, 1967, §307)
The neuter suffix reduces valence: *lahla* ‘throw away’ – *lahlela* ‘get lost’, *sēnya* ‘to damage’ – *sēnyeha* ‘to get damaged’ (Doke and Mofokeng, 1967, §308).
- (c) applicative (labeled *ap*, 2K occurrences) *-el-* (Doke and Mofokeng, 1967, §310)
The applicative suffix increases valence: *bōlela* ‘to say’ *bōlella* ‘to say to (s.o.)’ (Doke and Mofokeng, 1967, §310).
- (d) Perfective/Completive *-ella* (annotated *cl*, 66 occurrences) (Doke and Mofokeng, 1967, §336)
This does not actually change valence, but it is formally a reduplication of the applicative suffix (Doke and Mofokeng, 1967, §336), and as such its ordering behavior patterns with that of valence suffixes, in particular, it is placed before the passive suffix.⁵

⁵Example from the Demuth corpus: *u-neh-el-ets-w-a-ng ṭp.om2s-give-ap-cl-p-ṃin-wh* ‘What is it that you want passed to you?’.

(e) Reciprocal *-ana* (annotated rc, 103 times) (Doke and Mofokeng, 1967, §338)

This reduces valence: *rata* ‘to love’ – *ratana* ‘to love another’ (Doke and Mofokeng, 1967, §338).

Some of these suffixes can be stacked, e.g., see (Doke and Mofokeng, 1967, §345) for reversive+causative, and (Doke and Mofokeng, 1967, §314-315) for applicative suffixes applied to other valence affixes.⁶

Some other suffixes documented in the literature do not occur frequently or are not annotated in the corpus (e.g., the associative suffix (Doke and Mofokeng, 1967, §343)).

3. VOICE: passive *-w-* (labeled *p*, 1K occurrences) (Doke and Mofokeng, 1967, §300)

4. TENSE: tense (labeled *t̂*, 3K occurrences) .

The only tense suffix is the perfect affix *-il-*, which has a range of allomorphs depending on the preceding stem and valence/voice suffixes, if present (Doke and Mofokeng, 1967, §369), (Guma, 1971, p. 167). Common morphs in the Demuth corpus are *-il-* and *-its-*.

5. MOOD: Mood (labeled *m̂*, 37K occurrences)

In the Demuth corpus, the following mood endings are labeled (the analysis provided by Demuth (1992) is different from that provided by Doke and Mofokeng (1967), meaning the citations are only approximate):

(a) Imperative (labeled IMP) (Doke and Mofokeng, 1967, §386–387): singular (*-e*, labeled IMP) (Doke and Mofokeng, 1967, §386) and plural (*-ang*, labeled IMP.PL) (Doke and Mofokeng, 1967, §386).

Similar subjunctive SBJV1 *-e* (singular), *-eng* (plural).

(b) IND (*-a*, *-e*) and NEG (*-e*, *-a*) (Doke and Mofokeng, 1967, §394–421).

(c) subjunctive SBJV2 (*-e*, *-a*) (Doke and Mofokeng, 1967, §444–455)

6. Interrogative (labeled *wh*, 2K times) and relative (labeled *rl*, 857 times) markers *-ng*.

The interrogative marker *-ng* is a clitic form of *eng* ‘what’ according to (Guma, 1971, p. 168), (Doke and Mofokeng, 1967, §160, 320, 714); it is treated as a suffix in the Demuth corpus.

The relative marker *-ng* is affixed to verbs in relative clauses are marked with *-ng* (Doke and Mofokeng, 1967, §271, 793).

Examples from Demuth (1992):

Sbj.	Obj.	V	Val.	Voice	T.	M.
o		pheh			il	e (Thabo) cooked (food) (Demuth (1992) (15))
ke	e	f		uw		e (I) was given (the book) (Demuth (1992) (26c))
o		pheh	el			a (Thabo) cooks (food for Mpho) (Demuth (1992) (41))
o		pheh	el	w		a (Mpho) is being cooked (food) (Demuth (1992) (42))

⁶Example of reciprocal+applicative from Demuth corpus: *ba-arol-el-an-a sm2-t̂p.divide-ap-rc-m̂*in ‘Do they share?’

4.3 Experiment

Identifying underlying morphemes in Japanese In Japanese, we labeled suffixes for underlying morphemes with the aid of provided lemmatization. In most cases, underlying morphemes correspond to lemmas in the UD treebank. For the causative suffix, the treebank uses the lemmas *saseru* and *seru* depending on the verb stem. As passive and potential suffixes are formally identical for many verbs, they are not fully distinguished in the treebank annotation; we collapsed them into a single underlying morpheme labeled *Passive/Potential*. It corresponds to the lemmas *reru*, *rareru*, *eru*, *keru* in the treebank annotation.

Quantifying Prediction on the Phoneme Level In the main paper, we quantified prediction on the level of morphemes. We also repeated the experiments with prediction quantified on the level of phonemes.

For Japanese, we transliterated verb forms into syllabic Hiragana with the tagger Kytea (Neubig and Mori, 2010; Neubig et al., 2011), and then automatically phonemized these syllabic representations.

For Sesotho, we use the phonological transcription provided in the Demuth corpus. The Sesotho corpus has some cases of merged forms, where neighboring morphemes are merged and not segmented further. While we represented these as the corresponding sequence of underlying morphemes when modeling morpheme prediction, we ordered these merged phonemes according to the position that a grammar assigns to its first morpheme for modeling prediction on the phoneme level.

Estimating Predictability on Training Set In the main paper, we used the heldout set to estimate the memory-surprisal tradeoff when optimizing orders for AUC. We also repeated experiments using instead the training set. In this case, we did not apply smoothing; instead, we directly computed I_t for the empirical distribution given by the training corpus. We refer to this estimation method as the ‘naive’ estimator, because it directly applies the definition of I_t to the distribution defined by the n -gram counts in the training set.

Results Results for the memory-surprisal tradeoffs are shown in Figures (5–7). Accuracies on predicting orderings are shown in Figures (6–8). In the main paper, we report accuracies computed over all forms occurring in the corpus, counting each form by the number of times it occurs. This corresponds to the ‘Tokens’ results in Figures (6–8). Additionally, we also provide accuracies computed when counting each form only once, no matter how often it occurs; these are the ‘Types’ results. This method downweights high-frequency forms and upweights low-frequency forms. Results largely agree between the two methods, showing that results are not driven specifically by high-frequency forms. In Figures (6–8), we provide results both for optimizing on the heldout set as in the main paper, and for optimizing for the training set (‘Naive’). Results largely agree between the two methods.

References

- Bentz, C., Alikaniotis, D., Cysouw, M., and Ferrer-i Cancho, R. (2017). The entropy of words—learnability and expressivity across more than 1000 languages. *Entropy*, 19(6):275.
- Cover, T. M. and Thomas, J. (2006). *Elements of Information Theory*. John Wiley & Sons, Hoboken, NJ.
- Crutchfield, J. P. and Feldman, D. P. (2003). Regularities unseen, randomness observed: Levels of entropy convergence. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 13(1):25–54.

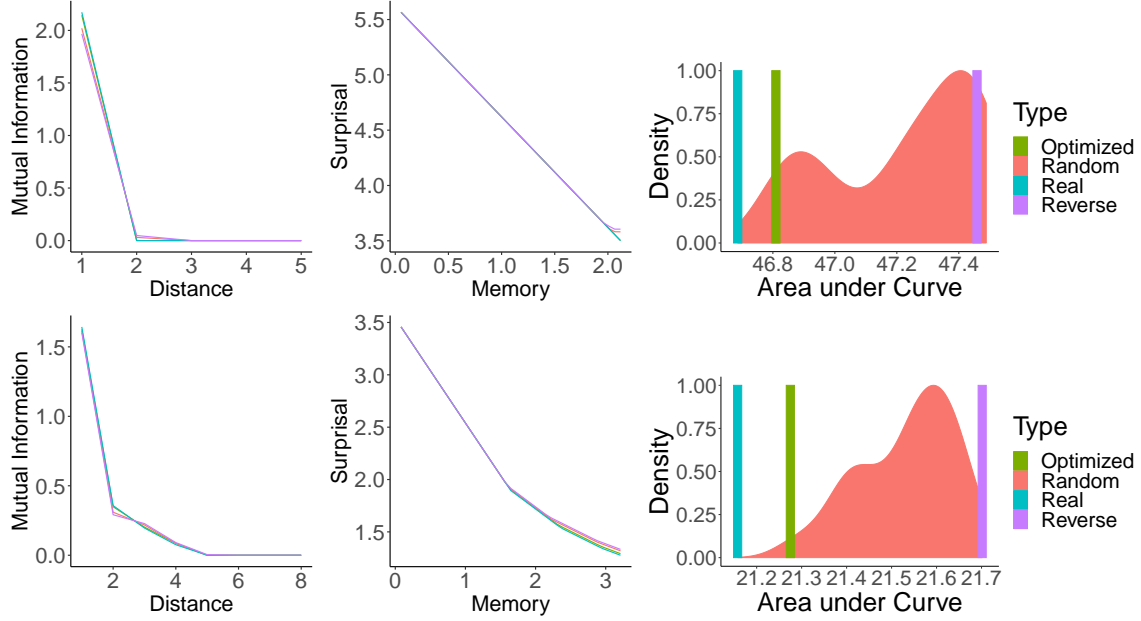


Figure 5: Japanese verb suffixes, measuring prediction on the level of morphemes (top) and phonemes (bottom), for real, random, approximately optimized, and reverse orderings. Left: I_t as a function of t . Center: Memory-surprisal tradeoff. Right: Areas under the curve for the memory-surprisal tradeoff.

			Pairs	Full
Tokens	Naive	Optimized for Phoneme Prediction	0.982 (SD 0.001)	0.979 (SD 0.001)
		Optimized for Morpheme Prediction	0.93 (SD 0.011)	0.919 (SD 0.009)
	Heldout	Optimized for Phoneme Prediction	0.963 (SD 0.006)	0.958 (SD 0.006)
		Optimized for Morpheme Prediction	0.953 (SD 0.011)	0.943 (SD 0.014)
	Random Baseline		0.496 (SD 0.269)	0.415 (SD 0.271)
Types	Naive	Optimized for Phoneme Prediction	0.974 (SD 0.002)	0.969 (SD 0.002)
		Optimized for Morpheme Prediction	0.903 (SD 0.015)	0.883 (SD 0.013)
	Heldout	Optimized for Phoneme Prediction	0.948 (SD 0.009)	0.938 (SD 0.009)
		Optimized for Morpheme Prediction	0.937 (SD 0.014)	0.921 (SD 0.017)
	Random Baseline		0.496 (SD 0.269)	0.415 (SD 0.271)

Figure 6: Accuracy of approximately optimized orderings, and of random baseline orderings, in predicting verb suffix order in Japanese. ‘Pairs’ denotes the rate of pairs of morphemes that are ordered correctly, and ‘Full’ denotes the rate of verb forms where order is predicted entirely correctly. We show means and standard deviations over different runs of the optimization algorithm (‘Optimized’), and over different random orderings (‘Random’). ‘Tokens’ results are obtained by counting each form by the number of occurrences in the data set; ‘Types’ results count each form only once. ‘Naive’ models are optimized for in-sample AUC, ‘Heldout’ models are optimized for heldout AUC. The figures in the main paper correspond to the *Heldout + Optimized for Morpheme Prediction* figures.

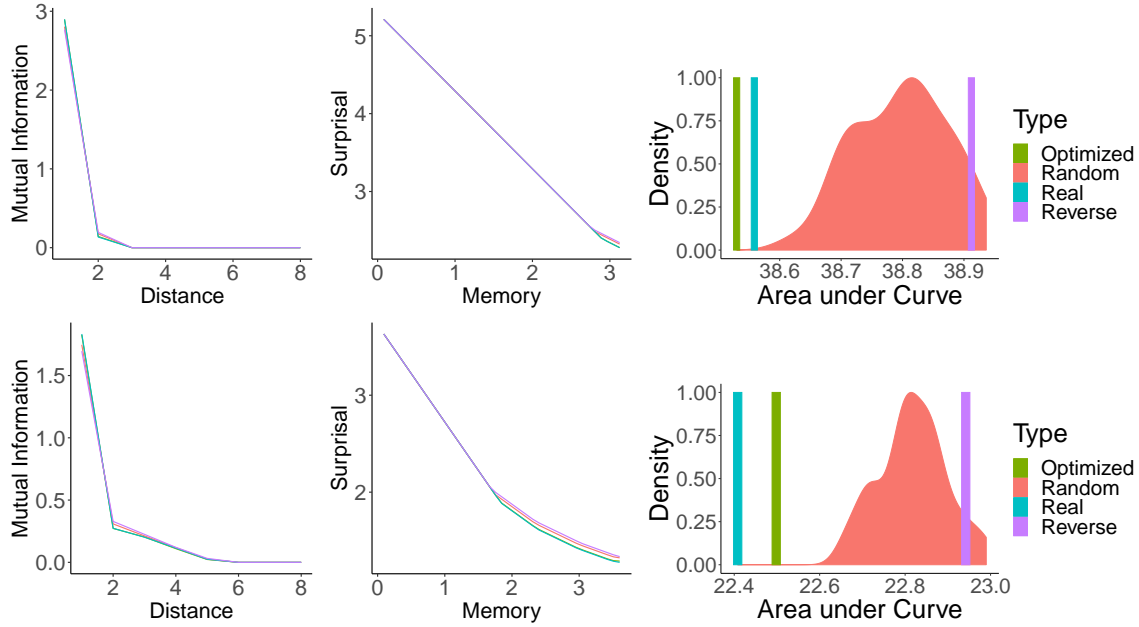


Figure 7: Sesotho verb affixes, measuring prediction on the level of morphemes (top) and phonemes (bottom), for real, random, approximately optimized, and reverse orderings. Left: I_t as a function of t . Center: Memory-surprisal tradeoff. Right: Areas under the curve for the memory-surprisal tradeoff.

Daniluk, M., Rocktäschel, T., Welbl, J., and Riedel, S. (2017). Frustratingly short attention spans in neural language modeling. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Debowski, L. (2011). Excess entropy in natural language: Present state and perspectives. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(3):037105.

Demberg, V., Keller, F., and Koller, A. (2013). Incremental, predictive parsing with psycholinguistically motivated tree-adjointing grammar. *Comput. Linguistics*, 39(4):1025–1066.

Demuth, K. (1992). Acquisition of sesotho. In *The cross-linguistic study of language acquisition*, pages 557–638. Lawrence Erlbaum Associates.

Doke, C. M. and Mofokeng, S. M. (1967). *Textbook of southern Sotho grammar*. Longmans.

Doob, J. L. (1953). *Stochastic processes*. New York Wiley.

Ebeling, W. and Pöschel, T. (1994). Entropy and Long-Range Correlations in Literary English. *Europhysics Letters (EPL)*, 26(4):241–246.

Goodman, J. (1999). Semiring parsing. *Comput. Linguistics*, 25(4):573–605.

Guma, S. M. (1971). *An outline structure of Southern Sotho*. Shuter and Shooter.

Hahn, M. and Futrell, R. (2019). Estimating predictive rate-distortion curves via neural variational inference. *Entropy*, 21(7):640.

				Prefixes		Suffixes	
				Pairs	Full	Pairs	Full
Tok.	Naive	Phon.	Opt.	0.985 (SD 0.0)	0.979 (SD 0.0)	0.989 (SD 0.0)	0.987 (SD 0.0)
			Rand.	0.361 (SD 0.307)	0.273 (SD 0.319)	0.431 (SD 0.198)	0.39 (SD 0.204)
		Morph.	Opt.	0.999 (SD 0.0)	0.998 (SD 0.0)	0.806 (SD 0.0)	0.723 (SD 0.0)
			Rand.	0.398 (SD 0.313)	0.303 (SD 0.319)	0.569 (SD 0.208)	0.511 (SD 0.228)
	Heldout	Phon.	Opt.	0.993 (SD 0.0)	0.989 (SD 0.0)	0.855 (SD 0.139)	0.836 (SD 0.152)
			Rand.	0.361 (SD 0.307)	0.273 (SD 0.319)	0.431 (SD 0.198)	0.39 (SD 0.204)
		Morph.	Opt.	0.99 (SD 0.0)	0.992 (SD 0.0)	0.756 (SD 0.012)	0.675 (SD 0.014)
			Rand.	0.398 (SD 0.313)	0.303 (SD 0.319)	0.569 (SD 0.208)	0.511 (SD 0.228)
Typ.	Naive	Phon.	Opt.	0.976 (SD 0.0)	0.966 (SD 0.0)	0.985 (SD 0.0)	0.98 (SD 0.0)
			Rand.	0.365 (SD 0.294)	0.267 (SD 0.296)	0.447 (SD 0.22)	0.398 (SD 0.235)
		Morph.	Opt.	0.997 (SD 0.0)	0.996 (SD 0.0)	0.844 (SD 0.0)	0.758 (SD 0.0)
			Rand.	0.405 (SD 0.308)	0.303 (SD 0.305)	0.546 (SD 0.197)	0.464 (SD 0.22)
	Heldout	Phon.	Opt.	0.988 (SD 0.0)	0.982 (SD 0.0)	0.871 (SD 0.118)	0.852 (SD 0.125)
			Rand.	0.365 (SD 0.294)	0.267 (SD 0.296)	0.447 (SD 0.22)	0.398 (SD 0.235)
		Morph.	Opt.	0.983 (SD 0.0)	0.986 (SD 0.0)	0.782 (SD 0.018)	0.697 (SD 0.02)
			Rand.	0.405 (SD 0.308)	0.303 (SD 0.305)	0.546 (SD 0.197)	0.464 (SD 0.22)

Figure 8: Accuracy of approximately optimized orderings, and of random baseline orderings, in predicting verb affix order in Sesotho. ‘Pairs’ denotes the rate of pairs of morphemes that are ordered correctly, and ‘Full’ denotes the rate of verb forms where order is predicted entirely correctly. We show means and standard deviations over different runs of the optimization algorithm (‘Opt.’), and over different random orderings (‘Random’). ‘Tokens’ results are obtained by counting each form by the number of occurrences in the data set; ‘Types’ results count each form only once. ‘Naive’ models are optimized for in-sample AUC on the training set, ‘Heldout’ models are optimized for heldout AUC.

	Real	Optimized
	Stem	Stem
1	suru	future
2	causative	desiderative
3	passive/potential	causative
4	desiderative	suru
5	politeness	passive/potential
6	negation	politeness
7	future	negation
	past	nonfinite
	nonfinite	past

Figure 9: Comparing order of Japanese affixes in the observed orders (left) and according to an approximately optimized grammar (right), optimized for AUC on the *training* set.

	Real	Optimized
1	Subject (relative)	Subject
	Subject	Subject (relative)
2	Negation	Negation
3	Tense/aspect	Tense/aspect
4	Object	Object
	Stem	Stem
1	Reversive	Reversive
2	Causative	Reciprocal
	Neuter	Causative
	Applicative	Neuter
	Reciprocal	Applicative
3	Passive	Passive
4	Tense/aspect	Tense/aspect
5	Mood	Interrogative
6	Interrogative	Relative
	Relative	Mood

Figure 10: Comparing order of Sesotho affixes in the observed orders (left) and according to an approximately optimized grammar (right), optimized for AUC on the *training* set. Note that order was separately optimized for prefixes and suffixes.

Hasegawa, Y. (2014). *Japanese: A linguistic introduction*. Cambridge University Press.

Hochreiter, S. and Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Jelinek, F. and Lafferty, J. D. (1991). Computation of the probability of initial substring generation by stochastic context-free grammars. *Comput. Linguistics*, 17(3):315–323.

Kaiser, S., Ichikawa, Y., Kobayashi, N., and Yamamoto, H. (2013). *Japanese: A comprehensive grammar*. Routledge.

Kim, Y., Dyer, C., and Rush, A. M. (2019). Compound probabilistic context-free grammars for grammar induction. In Korhonen, A., Traum, D. R., and Màrquez, L., editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2369–2385. Association for Computational Linguistics.

Kneser, R. and Ney, H. (1995). Improved backing-off for m-gram language modeling. In *Acoustics, Speech, and Signal Processing, 1995. ICASSP-95., 1995 International Conference on*, volume 1, pages 181–184. IEEE.

Lewis, R. L. and Vasishth, S. (2005). An activation-based model of sentence processing as skilled memory retrieval. *Cognitive Science*, 29(3):375–419.

Lin, H. W. and Tegmark, M. (2017). Critical Behavior in Physics and Probabilistic Formal Languages. *Entropy*, 19(7):299.

McElree, B. (2000). Sentence comprehension is mediated by content-addressable memory structures. *Journal of psycholinguistic research*, 29(2):111–123.

- Mikolov, T., Karafiát, M., Burget, L., Ěernocký, J., and Khudanpur, Sanjeev (2010). Recurrent neural network based language model. In *Proceedings of INTERSPEECH*.
- Nederhof, M. and Satta, G. (2011). Computation of infix probabilities for probabilistic context-free grammars. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing, EMNLP 2011, 27-31 July 2011, John McIntyre Conference Centre, Edinburgh, UK, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1213–1221. ACL.
- Neubig, G. and Mori, S. (2010). Word-based partial annotation for efficient corpus construction. In Calzolari, N., Choukri, K., Maegaard, B., Mariani, J., Odijk, J., Piperidis, S., Rosner, M., and Tapias, D., editors, *Proceedings of the International Conference on Language Resources and Evaluation, LREC 2010, 17-23 May 2010, Valletta, Malta*. European Language Resources Association.
- Neubig, G., Nakata, Y., and Mori, S. (2011). Pointwise prediction for robust, adaptable japanese morphological analysis. In *The 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies, Proceedings of the Conference, 19-24 June, 2011, Portland, Oregon, USA - Short Papers*, pages 529–533. The Association for Computer Linguistics.
- Nicenboim, B. and Vasishth, S. (2018). Models of retrieval in sentence comprehension: A computational evaluation using bayesian hierarchical modeling. *Journal of Memory and Language*, 99:1–34.
- Petrov, S. and Klein, D. (2007). Learning and inference for hierarchically split pcfgs. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence, July 22-26, 2007, Vancouver, British Columbia, Canada*, pages 1663–1666. AAAI Press.
- Roark, B. (2001). Probabilistic top-down parsing and language modeling. *Comput. Linguistics*, 27(2):249–276.
- Schadeberg, T. (2003). Derivation. In *The Bantu Languages, edited by D. Nurse & G. Philippson*, 71-89. Routledge, London.
- Schijndel, M. V., Exley, A., and Schuler, W. (2013). A model of language processing as hierarchic sequential prediction. *topiCS*, 5(3):522–540.
- Shannon, C. E. (1951). Prediction and entropy of printed english. *Bell system technical journal*, 30(1):50–64.
- Snoek, J., Larochelle, H., and Adams, R. P. (2012). Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959.
- Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.
- Stolcke, A. (1995). An efficient probabilistic context-free parsing algorithm that computes prefix probabilities. *Comput. Linguistics*, 21(2):165–201.
- Takahashi, S. and Tanaka-Ishii, K. (2018). Cross entropy of neural language models at infinity—a new bound of the entropy rate. *Entropy*, 20(11):839.

- Vaccari, O. and Vaccari, E. E. (1938). *Complete course of Japanese conversation-grammar*. Maruzen in Komm.
- Vasishth, S., Nicenboim, B., Engelmann, F., and Burchert, F. (2019). Computational models of retrieval processes in sentence processing. *Trends in Cognitive Sciences*.
- Xie, Z., Wang, S. I., Li, J., Lévy, D., Nie, A., Jurafsky, D., and Ng, A. Y. (2017). Data noising as smoothing in neural network language models. *arXiv preprint arXiv:1703.02573*.