The emergence of grammatical structure from inter-predictability

John Mansfield and Charles Kemp
University of Melbourne

To appear in *A Festschrift for Jane Simpson*

Abstract
Recent research has shown that words or morphemes that are closer to each other in linear order tend to have higher statistical inter-predictability, measured as mutual information. We offer an explanation for this in terms of holistic chunking of inter-predictable symbols, which provides an efficiency gain in the retrieval of stored symbols to encode a message. Inter-predictable chunking then interacts with structural priming to produce the schematic linear structures that are characteristic of both syntax and morphology. We thus argue that predictability and efficiency play a key role in the emergence of grammatical structure, going beyond previous information-theoretic analyses of natural language. In this paper we articulate some fundamental principles of chunking and linearisation, and use a simple computational implementation to show that these are sufficient to produce natural-language-like structures, using NP-internal ordering as a case study.

1. Introduction

How did grammatical structures evolve to be the way they are? Presumably they have developed slowly over thousands of years of human language learning and use, but what forces in particular have shaped outcomes such as a noun phrase schema, Dem-Num-Adj-N? More importantly, we might ask how human language has arrived at its global distribution such that Dem-Num-Adj-N is common to many distinct lineages, Dem-Num-N-Adj somewhat less common, and N-Num-Dem-Adj occurs hardly at all (Dryer, 2018).

There are two main approaches to explaining these phenomena: theories that involve a fixed underlying syntactic structure, around which specific languages ebb and flow (e.g. Cinque & Rizzi, 2009; Chomsky & Lasnik, 2015 [1993]), versus theories where there is no fixed underlying structure, only flow (e.g. Hopper, 1987; Bybee, 2001; Hopper & Traugott, 2003). This paper belongs firmly in the second camp. Although we cannot demonstrate the actual evolutionary development of Dem-Num-Adj-N, in English or any other language, we propose some fundamental principles that we believe to have played a key role in the evolution of such structures. We believe that these principles are active in human cognition when individuals acquire and process language, but modelling actual acquisition and processing are beyond our reach in this study. Instead, we demonstrate the operation of our principles in a relatively simple computational implementation. We firstly articulate the core principles, and secondly present a preliminary implementation using artifical language data. A follow-up study will apply a similar model to natural corpus data (Mansfield & Kemp, in prep.).

This paper is a tribute to Professor Jane Simpson, who may or may not agree with any of its conclusions. One of Professor Simpson's major contributions to grammatical theory is her study of 'templatic' structure in morphology (Simpson & Withgott, 1986), which we here integrate into a more general proposal about linear grammatical schemas, encompassing both morphology and syntax. Simpson and Withgott pointed out a range of idiosyncratic morphological constructions that defy the main tenets of phrase-structure grammar, and should therefore be treated as an alternative type of grammatical computation. In this paper we do not dwell on the question of morpheme-specific idiosyncracies, but instead take the general idea of linear schemas as a point of departure for a model of how some morphosyntactic structures can be explained without any recourse to a fixed hierarchy of grammatical categories. In fact, while templates or linear schemas are often regarded as 'arbitrary' or 'stipulative' analyses that fall outside of principled linguistic theory, we will argue that on the contrary, linear schemas can be explained based on very sound principles of linguistic cognition.

Recent research has shown striking correspondences between linear proximity of words or morphemes, and their statistical inter-predictability, measured as mutual information (Futrell, 2019). For example, a grammatical category order such as Dem-Num-Adj-N, familiar from languages such as Ket (1), corresponds with corpus data showing that adjectives are the most inter-predictable with nouns, numbers less so, and demonstratives least of all (Culbertson et al., 2020).

(1) *Ket*
kin'e qo· aqta dɛˀŋ
DEM ten good people
'these ten good people' (Rijkhoff, 2004, p. 126)

This paper proposes an explanation for why linear proximity and inter-predictability should be correlated, with holistic retrieval of complex symbols being the key connection. We test this proposal by implementing a computational model and showing that it generates similar grammatical sequences to those found in natural languages, starting from input data that has no intrinsic ordering. The model is not a realistic simulation of any human language, since there is no situation in which a language user must devise linear sequencing naively. Our model begins with completely random ordering, while users of natural language are exposed to particular sequencing patterns, which they reproduce with occasional modifications. Rather than modelling actual language use or acquisition, our more modest goal in this paper is to show that inter-predictability can *in principle* account for grammatical linear ordering, and to propose a novel way of associating these phenomena, via efficient chunking.

Since our model does not reflect many properties of actual human language, rather than call it a 'language' or a 'grammar', we refer to it more modestly as an 'encoder'. The encoder does not require any intrinsic ordering principles that reference specific grammatical categories, but instead achieves natural-language-like sequences based on independently evidenced principles of linguistic cognition. We propose three fundamental principles that drive linear ordering (these will be explained in more detail below):

(I) **Chunking**: To reduce the number of symbols processed, any group of symbols that is highly inter-predictable can be stored and retrieved as a single complex symbol;

(II) **Adjacency**: When a message is output as a linear sequence, complex symbols are preferably linearised with their sub-parts adjacent to one another;

(III) **Structural priming**: When a message is output as a linear sequence, associative memory residues from previous messages make the grammatical category sequence of the new message more likely to match those of previous messages.

If a grammatical structure can be shown to emerge from these principles, then there is no need to propose an explanation involving a fixed underlying syntax of grammatical categories. Our claim is not that *all* grammatical structure can be explained in this way – for example, it seems likely that basic constituent order (SOV etc) requires other principles. But we propose that many linear structures are at least partially explained by principles I–III. In this study we support this claim by applying the encoder to NP-internal categories {N, Adj, Num, Dem}. The encoder does a good job of generating NP-internal orderings that bear a striking resemblance to natural languages, but there is nothing about the system that is specific to these grammatical categories, and therefore it should be applicable to a range of other linear structures.

Before describing the encoder, we briefly reflect on how Professor Simpson's work provokes consideration of linear versus hierarchical structure (§2) and summarise recent research that demonstrates the widespread correspondence of inter-predictability with linear ordering (§3).

2. Linear versus hierarchical, stipulative versus principled

Simpson & Withgott identify a type of grammatical structure that is better analysed as 'linear arrays of morphemes', rather than hierarchical structure (Simpson & Withgott, 1986, p. 156; see also Perlmutter, 1971). They use the term 'templates' for the schematic generalisations that can be made over such morpheme arrays. A template is fully schematic if it refers only to grammatical categories, e.g. TAM-Subj-Obj, where all morphemes belonging to these categories are linearised in that way. But templates may also be 'partially schematic', where some positions are filled by grammatical categories and others by specific morphemes, or by heterogeneous sets of morphemes.

Partially schematic templates allow for idiosyncratic dependencies between morphemes, in a way that does not reflect classic hierarchical phrase-structure representations. Simpson & Withgott present a range of examples from Warumungu, Warlpiri and French. For example, Warumungu has pronominal clitic clusters with idiosyncratic ordering depending on exactly which subject/object combinations are involved (Simpson & Withgott, 1986, p. 161).[1] Subsequent research has demonstrated many other examples of idiosyncratic sequencing in morphology (e.g. Stump, 1997; Nordlinger, 2010b), including paradigmatic inconsistencies that might, for example, place 1SG.S in one linear position, but 2SG.S in a quite different position (Crysmann & Bonami, 2016; Mansfield et al., 2022).

By contrast, hierarchical phrase-structure grammar is a fully schematic model of grammatical category relations: the rules generalise across all members of grammatical categories, as opposed to say, having lexically specific phrasal rules. This produces a fixed underlying structure of grammatical categories, and in some versions this hierarchy is fixed for all human languages (Chomsky & Lasnik 2015 [1993]). Since the clitic clusters in Warumungu instead involve idiosyncratic dependencies between morphemes, these are claimed to belong to another type of grammar – the template. The structure of templates is considered to be language-specific and stipulative – i.e. it is free from the constraints of general grammatical principles (Good 2016). By association, *linear sequences* are often seen as stipulative and language-specific, in contrast to hierarchical structure which is principled and universal. Indeed, some influential universalist models take hierarchy to its logical limit, by proposing that all structure is strictly binary-branching (Kayne 1981, 1994). The slogan 'structures not strings' captures the view that hierarchical grouping is the proper subject of syntactic theory, while linear sequences are superficial and/or unimportant (Everaert et al., 2015; Kulmizev & Nivre, 2022 inter alia).

In contrast to the widespread view that linear schemas are stipulative and unprincipled, in this paper we will argue that linear grammatical schemas are based on very robust, general principles. Linear schemas found in particular languages are not failures of linguistic theory, but rather can be incorporated into a stochastic model of symbolic encoding and linearisation that predicts constrained variation in grammatical structure. The principles that we will evoke involve efficient processing and associative memory, both of which are extensively evidenced in human language use. However, we do not claim that *all* grammatical structure is linear. Hierarchical structure is clearly motivated for structures with unbounded recursion (Chomsky 1957), and perhaps even more essentially, those parts of grammar that involve recurrent expandable nodes, such as the multiple NP positions in clause structure (Wells, 1947). Rather, our claim is that hierarchical grouping should be proposed only where it is clearly motivated, while other

[1] This could be taken to imply that *=ngki* is an 'inverse' marker, but in fact it only appears in this specific clitic cluster, which undermines any analysis as a 'recurrent partial' with an independent meaning.

structures can be sufficiently modelled as linear arrays.[2] This is similar to the position argued by Culicover & Jackendoff (2005), who note that hierarchical structure adds complexity to syntactic representations and to derivational processes, and should therefore be posited only as necessary.

There is an interesting parallel between our proposal and theories of hierarchical syntax, since our model does in fact involve recursive chunking of linguistic symbols (see details below). But the important difference is that our hierarchical structures are a format for storing and retrieving specific linguistic symbols, as opposed to phrase-structure grammr, which involves a hierarchy of grammatical categories. We return to this difference in the discussion section below.

3. Inter-predictability and linear proximity

Several recent studies have revealed a striking relationship between linear proximity and statistical inter-predictability, evidenced in both syntax and morphology. Futrell, Hahn and colleagues have identified various correlations, both with respect to specific constructions such as adjective stacking (e.g. *beautiful green shirt* versus ?*green beautiful shirt*) (Hahn et al., 2018), and more widely with respect to the proximity of words in phrase structure, and the proximity of affixes to the stem in word structure (Futrell, 2019; Hahn et al., 2021, 2022). Pairs of morphemes (either words or affixes) are INTER-PREDICTABLE when they tend to co-occur in the same phrases, rather than independently. For example, *man* + *big* tend to co-occur, as do *house* + *large*, showing that the occurrence of these nouns and adjectives are not statistically independent (Biber et al., 1998, p. 46). The crucial observation is that inter-predictable morphemes tend to occur close together in linear sequences, a principle that Futrell labels INFORMATION LOCALITY (Futrell, 2019). Futrell, Hahn and colleagues propose an explanation for information locality based on the idea that predictability helps us process the incoming linguistic signal (Hale, 2001; Levy, 2008), but our memory of contextual predictors decays over time. Therefore, the benefit of predictive processing is maximised when highly inter-predictable symbols have minimal gaps between them (Futrell, 2019; Hahn et al., 2021). They call this the 'memory–surprisal tradeoff'. Other relevant strands of theory include dependency locality (e.g. Hawkins, 2004; Futrell et al., 2020; Jing et al., 2022), and uniform information density (Jaeger, 2010; Jaeger & Buz, 2017), which may or may not align with information locality.

In this study we propose a different explanation for information locality, where storage and retrieval of complex symbols is the cognitive mechanism responsible. Let us call this new proposal INFORMATIONAL CHUNKING. The key insight of informational chunking is anticipated in some earlier studies of English morphology (Hay, 2002; Plag & Baayen, 2009), arguing that stem-affix adjacency is explained by certain combinations

[2] More specifically, if we assume that grammar has both linear schemas and hierarchical nesting, this implies a model of linear schemas in which some positions are themselves filled by schemas. This is not mentioned in Simpson & Withgott's proposal, but is later suggested by Good, who calls these 'elastic' positions (Good, 2016, p. 57).

being more likely to be holistically processed. This holistic processing is in turn related to statistical association measures revealed in corpus data. We build on these earlier studies by connecting these findings to information theory, and extending the reach of the theory beyond morphology to grammatical linearisation in general. Alongside the memory–surprisal tradeoff, dependency locality, and uniform information density, our proposal reflects a blossoming of research on the question of how functional constraints and cognitive biases shape grammatical structure. Comparing the merits and predictions of these theories would be of great interest, but we must leave such a comparison to future work.

One of the recent studies demonstrating information locality is Culbertson and colleagues' (2020) cross-linguistic investigation of {N, Adj, Num, Dem} sequencing in noun phrases (Culbertson et al., 2020). This is interesting because there is disagreement in the literature about whether noun phrases should be treated as linear or hierarchical structures (e.g. Culicover & Jackendoff, 2005; Alexiadou et al., 2008; see further references below). While the elements {N, Adj, Num, Dem} show a diverse range of sequences, most languages conform to a general constraint, whereby linear proximity to the noun is ranked as Adj ≥ Num ≥ Dem (the adjective must be at least as close as the number, and the number must be at least as close as the demonstrative). Of the 24 possible permutations of the categories {N, Adj, Num, Dem}, there are eight orderings that satisfy this ranking constraint, and these account for 83% of languages in a typological sample of 576 languages (Dryer, 2018, p. 799). The authors calculate inter-predictability statistics from syntactically annotated corpora and find that the ranking stated above is reflected in the degree to which each of these modifier classes is inter-predictable with nouns, and this is true for all 24 languages tested (Culbertson et al., 2020, p. 697). This reveals a striking example of information locality, i.e. a correlation between linear proximity and inter-predictability.

Information locality in noun phrases connects with an earlier suggestion by Dryer (2009, p. 197), who proposes that word classes are closer to the noun if they 'denote more inherent properties of the referent'. Adjectival properties such as colour and size tend to be permanent properties of referents, and thus have a predictable relationship to noun labels, whereas numerosities and deictic markers are ephemeral (most entities can freely appear in different numbers, or different relations to the speaker/addressee), and thus number and demonstrative have much less statistical inter-predictability with noun labels. Thus our experience of the world, and the semantic basis of noun-phrase categories, seems to provide a plausible account of why some word classes have greater statistical inter-predictability than others (Culbertson et al., 2020, p. 702). Nonetheless, in our view it is the statistical aspect of this that is the proximal cause of grammatical linearisation, and thus the main focus of this study.

Culbertson and colleagues interpret NP linear schemas as evidence of an underlying nested conceptual representation [Dem [Num [Adj [N] Adj] Num] Dem] (see also Rijkhoff, 2004, p. 218). The nested conceptual representation generates isomorphic word orders, explaining the strong bias towards certain NP-internal sequences across languages (Culbertson et al., 2020, p. 709). This explanation is compatible with hierarchical syntactic accounts in which the linear ordering of NP-internal categories is modelled as an underlying structure along the lines of Figure 1 (Alexiadou et al., 2008; e.g. Cinque, 2005).

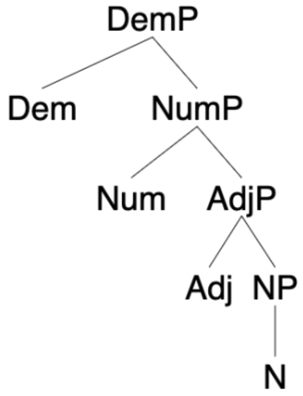

Figure 1. Hierarchical syntax tree for the elements {N, Adj, Num, Dem}.

In this paper, we take Culbertson and colleagues' NP findings as the basis for a model of how linear sequences of grammatical categories can be explained by inter-predictability. But rather than treating word order as mediated by a fixed underlying hierarchical structure, we instead propose a mechanism that more directly generates word ordering from inter-predictability. A position closer to ours is held by Dryer (2009, 2018), who treats the NP as a linear structure, and discusses a range of functional motivations for linearisation, including the abovementioned 'inherent properties' theory. But our proposal does not even strictly require a linear schema (though we don't rule it out as an efficient abstraction); instead, our linearisation procedure works directly by priming from previous exemplars (cf. Ambridge, 2020a, 2020b). In the implementation described below, messages are simply linear sequences of symbols, and consistent grammatical category sequences are an emergent property of inter-predictability of symbols, together with associative memory.

4. Generating {N, Adj, Num, Dem} ordering from cognitive and communicative principles

Our encoder begins with no inherent ordering of grammatical categories, and develops schematic ordering based on independently evidenced cognitive and communicative principles. Repeated from above, these are:

(I) **Chunking**: To reduce the number of symbols processed, any group of symbols that is highly inter-predictable can be stored and retrieved as a single complex symbol;

(II) **Adjacency**: When a message is output as a linear sequence, complex symbols are preferably linearised with their sub-parts adjacent to one another;

(III) **Structural priming**: When a message is output as a linear sequence, associative memory residues from previous messages make the grammatical category sequence of the new message more likely to match those of previous messages.

Each of these is a sound working assumption, supported by extensive previous research. **Chunking** follows from information-theoretic research on effort reduction (e.g. Shannon, 1948; Zipf, 1949; Aylett & Turk, 2006; Grünwald, 2007; Levshina, 2022), though as we explain below, the application here refers to effort-reduction in the retrieval of symbols, rather than their external articulation (Seržant & Moroz, 2022). **Adjacency** is obvious enough to be tacitly assumed in psycholinguistic research on holistically-memorised complex symbols. In most or all studies, the stored complex symbols are output as contiguous units, and this goes equally for memorisation of morphologically complex words (e.g. Baayen & Schreuder, 2003; Marslen-Wilson, 2007; Kuperman et al., 2010), or multi-word phrases (e.g. Arnon & Snider, 2010; Pijpops et al., 2018; Contreras Kallens & Christiansen, 2022). Adjacency is also central, though somewhat more flexibly operationalised, in the large body of research on corpus collocations (e.g. Sinclair, 1991). **Structural priming** and persistence is extensively evidenced in language processing (e.g. Szmrecsanyi, 2006; Van Gompel & Arai, 2018). More specifically with respect to the ordering of grammatical categories, it is supported by evidence for category-based priming effects in selecting among variable word orders for a sentence (Hartsuiker et al., 1999; Fukumura & Zhang, 2023) and similar effects in morphological sequences (Mansfield et al., 2020, 2022). Consistent linear positioning of elements from the same grammatical category is tacitly assumed in most theories of grammar, but is stated explicitly by Dik (1997) as the PRINCIPLE OF FUNCTIONAL STABILITY, and is treated as a central principle by Hoey (2005).

We will show that a model implementing these principles produces very similar distributions of {N, Adj, Num, Dem} sequences to those observed in natural languages. We will also see how the relative weighting of these principles affects the emergence of grammatical ordering, and the important distinctive roles played by each principle. This suggests that at least some of the linear grammatical schemas observed in natural languages need not be seen as 'stipulative', but on the contrary can be motivated by fundamental, well-evidenced principles. In the course of explaining the model and illustrating its output, potential interpretations in terms of human language cognition will become apparent. The model is implemented as a single Python script of modest complexity (< 500 lines of code), which is available at the code repository for this study.[3]

4.1. Methodological preliminaries

The model described here is a simple 'message encoder', a proof-of-concept of using artificial language data. A follow-up study will present more complex implementations

[3] https://zenodo.org/record/7601752

using natural corpus data (Mansfield & Kemp, in prep.). Some features here are deliberately simplified, allowing us to focus on the main question of how inter-predictability of symbols might give rise to ordering.

The input data used for the encoder is an artificially generated dataset of co-occurring atomic concepts, assigned to categories {N, Adj, Num, Dem}. These form a set of 'messages', where each message contains one element from each grammatical category, in unordered sets such as {DOGS, BROWN, THREE, THESE} and {CATS, BLACK, FOUR, THOSE}. These sets of concepts are the semantic content of the messages, and the encoder builds a linear string of symbols to encode the content of each message.

Our message set is generated using a stochastic process, designed so that the inter-predictability of the grammatical categories is similar to that found in Culbertson et al (2020), namely: N;Adj $\geq$ N;Num $\geq$ N;Dem. Following Culbertson and colleagues, we formulate inter-predictability as mutual information (MI), which can be calculated for each specific combination of symbols, or averaged across each modifier category Adj, Num, Dem. We generated one set of approximately 5000 training messages, which is used by the encoder to estimate the pointwise mutual information (PMI) of each symbol combination, and a second list of 500 test messages, which the encoder must linearise. (The exact numbers of messages are variable due to the stochastic nature of the generation process.) The data has more possibilities for nouns, adjectives and numbers, since these are large open classes, and much more constrained possibilities for demonstratives, since this is a small closed class. There are eight nominal concepts DOGS, CATS, RABBITS, BIRDS, CHAIRS, TABLES, CARS, WHEELS; eight adjectival concepts BROWN, BLACK, WHITE, GREY, BIG, SMALL, OLD, NEW; eight numerical concepts TWO–NINE; and two deictic concepts THESE, THOSE. For technical details of how the datasets are created, see the script `generate-MI-distros.py` at the code repository.

The only important property of these datasets is that they reflect the kind of inter-predictability relations found in NPs in natural language corpora. Over 10 runs of the generator, the mean MI figures and standard deviations for each modifier category are: PMI(N;Adj) $\mu = 0.77$, $\sigma = 0.13$; PMI(N;Num) $\mu = 0.33$, $\sigma = 0.06$; PMI(N;Dem) $\mu < 0.01$, $\sigma < 0.01$. The absolute values of these figures are smaller than those reported for natural language corpora by Culbertson et al (2020: 705), but this is simply an effect of the smaller sets of distinct symbols. In any case, the only important thing for our model is the ranking of the grammatical categories.

Aside from inter-predictability ranking, the messages we generate are very unlike natural language, because they always consist of the categories {N, Adj, Num, Dem}. Obviously, humans usually use just a subset of these conceptual categories to form a referential expression, and the consistent use of all four types was implemented purely as a way to simplify the model. Future versions extend the model to allow the more flexible message content of natural corpora (Mansfield & Kemp, in prep.).

4.2. Inter-predictable symbols and the efficient communication trade-off

Before commencing the proper 'encoder' phase, there is a preparatory phase in which all atomic concepts are stored in memory as linguistic symbols, such that for each concept e.g. CAT, there is an associated symbol *cat*. More importantly, this memorisation process represents the statistical inter-predictabilities between all symbols, as calculated from the ~5000 messages in the training data. Once all symbols and their inter-predictabilities are stored in memory, the encoder proper works through the ~500 messages in the test data, using two distinct steps to encode each as a linear string: firstly, symbols required for expressing the concepts are retrieved from memory, either as simple or complex symbols; secondly, the resulting set of simple and complex symbols is output in a linear sequence. In this section we describe the preparatory phase of symbolic memorisation, and the symbolic retrieval process, which we call 'chunking'.

As mentioned above, the symbolic memorisation phase reads all the messages in the training data, and uses these to calculated inter-predictability for all associated symbols. This is formulated as pointwise mutual information (PMI) for each specific symbol combination, e.g. E.g. PMI(*cats*;*black*) = 1.26, or PMI(*dogs*;*three*) = 0.31. The use of PMI has a long tradition as an association measure in corpus linguistics (e.g. Church & Hanks, 1990). PMI is typically measured between pairs of variables, but it can also be applied recursively to calculate the inter-predictability between a binary combination and a third simple variable, e.g. PMI((*cats, black*); *three*), or similarly between a ternary combination and a simple fourth variable.[4]

Once all symbols have been memorised and their inter-predictabilities calculated, the encoder works through the ~500 messages in the test data, encoding each into a linear message sequence. As outlined above, the semantic content of each message includes nominal, adjectival, numeric and deictic concepts, for example {CATS, BLACK, THREE, THESE}. The encoder retrieves the associated symbols from memory, either as simple or complex symbols. For example, a set of symbols might be retrieved as {(*cats, black*), *three, these*}, i.e. with one binary chunked symbol and two simple symbols. We represent chunks with round brackets, and the complete set of retrieved symbols with curly brackets. Retrieving a chunked complex symbol reduces the number of symbols that need to be retrieved, and this can be interpreted as 'effort reduction'. One might think of this like retrieving clothes from a cupboard before getting dressed: if a particular belt tends to go with a particular pair of jeans, it may be more efficient to store and retrieve these already conjoined; a pair of socks that always go together should be stored and retrieved as a single bunched-up unit.

The storage of complex chunks does not prevent simple symbols from being accessed individually: for example, (*cats, black*) does not delete *cats* or *black* from memory. We also assume that the semantics and grammatical category membership for

[4] We might expect these multivariate PMIs to be less important than binary PMIs, but the encoder nonetheless calculates ternary and quaternary PMIs, since human language does show some effects of holistic storage for larger complex symbols (e.g. Arnon & Snider, 2010).

parts of complex symbols remains identical with simple symbols, setting aside the fact that in natural language, complex symbols may in some cases become semantically or grammatically dissociated from their parts over time (Brinton & Traugott, 2005). Thus the storage of complex symbols adds to the complexity and size of our lexical storage, as a trade-off against the benefit of more efficient retrieval (Wray, 2002, 2017). This type of efficiency reflects the same basic principles as the concept of chunking used in research on linguistic production planning (Blumenthal-Dramé et al., 2017), and theories of the mental lexicon (e.g. ten Hacken, 2019).

The connection we draw between efficient retrieval and mutual information is inspired by previous information-theoretic work on language processing (Gibson et al., 2019). However, while most information-theoretic work focuses on the number of symbols used in message *transmission* (e.g. articulatory reduction or word length), the proposal here applies the same mathematical principles to efficient retrieval of symbols from a lexicon (cf. Seržant & Moroz, 2022). In our model, chunking is based on the PMI of symbol combinations, which we call INFORMATIONAL CHUNKING. Since natural language exhibits pervasive variability, we model our chunking as a variable process, using a random error term drawn from a Gaussian distribution. The PMI of a symbol combination, plus a random error term, is tested against a chunking threshold. For example, the chunk threshold can be set at 1 bit, and the encoder might retrieve the concepts {CAT, BLACK} with PMI(*cats*;*black*) = 1.26, and a random error ε = -0.04, which would yield a result of 1.22 and thus pass the threshold for a complex chunk.

The encoder begins by testing binary combinations, potentially returning binary chunks like (*cats, black*); but the process is recursive, testing whether chunks can be the input to further chunking such as ((*cats, black*), *three*) or ((*black*, *three*), *cats*). At the maximum, all four symbols can be chunked into a single complex symbol, such as (((*cats, black*), *three*), *these*); however, PMI naturally tends to reduce as symbols grow more complex, so more complex symbols occur less frequently.

If it is easier to retrieve fewer symbols, we might wonder why not chunk the semantic content of every message as a single complex symbol? Intuitively, the problem here would be the exponential proliferation of an enormous range of complex symbols that need to be accurately retrieved. This implies an additional principle, which we did not highlight above as we take it to be an uncontroversial *a priori*: that there is some upper limit on the range of distinct symbols that can be stored and retrieved. Constrained by the symbol proliferation problem, the encoder should perform chunking selectively, to gain savings on the symbol-count of individual messages, but also limit the range of complex symbols it retrieves. The degree of proliferation could be formulated simply as the number of distinct (simple or complex) symbols retrieved by the encoder over its entire repertoire of messages. However a more nuanced formulation is the *entropy* of symbol retrieval (Shannon, 1948), which reflects both the number of distinct symbols, and whether some particular symbols are more frequently retrieved (less entropy), or whether all symbols have close to equally frequent usage (more entropy). In information-theoretic terms, we can think of symbol retrieval as having a *channel capacity*, which limits the per-symbol entropy at which retrieval can accurately occur (Cover & Thomas, 2002). The encoder thus has conflicting constraints to reduce the number of symbols retrieved per message, while limiting the retrieval entropy per symbol. Formulating the problem in these terms is related to the idea of two part-codes from the Minimum Description Length literature, which balance the length of a set of encoded messages against the length of the 'code book' used for coding these messages (Grünwald, 2007).

Figure 2 illustrates the superior efficiency of PMI-based chunking, in comparison to purely random chunking based only on the Gaussian error term. Both versions were run using a range of chunking thresholds, reflected in the range of data points output by each method. The PMI-based method demonstrates superior efficiency, as represented by its outputs being closer to the bottom-left quadrant of the graph. This indicates that it achieves fewer symbols per message, at a lower encoder entropy. PMI-based chunking generates a smaller set of complex symbols, such as (*cats, black*), which are frequently repeated. By contrast random chunking generates a large range of complex symbols, many of which appear only once. The latter scenario results in higher encoder entropy.

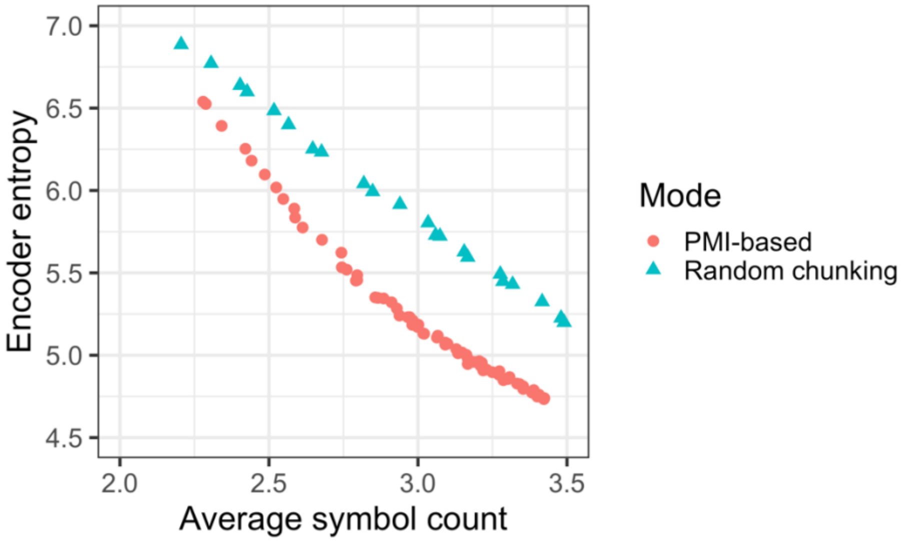

Figure 2. Trade-off between symbol count per message and encoder entropy, illustrating results from PMI-based chunking versus an encoder where chunking is purely random.

4.3. Linearisation via contingity and consistency parameters

The second step of string encoding involves linearisation of the retrieved symbols. As mentioned above, each set of symbols begins with no intrinsic ordering, but consistent grammatical category sequences emerge from the application of our two remaining fundamental principles: (II) complex symbols are preferentially transmitted in a contiguous fashion; and (III) previously used grammatical category sequences are preferentially re-used. Each of these principles is implemented as a parameter that can be applied at various weights. Our model therefore has two 'linearisation' parameters, contiguity and consistency, in addition to the chunking threshold parameter described above.

Once the symbolic content of a message has been retrieved, the set of simple and complex symbols are linearised to perform message transmission. Like chunking, sequencing is a fundamentally variable process. In the encoder's initial state, all the 24 possible permutations of the categories {N, Adj, Num, Dem} are equally probable as transmission sequences; but on each message transmission, these probabilities are modified by the two linearisation parameters.

Firstly, if the message includes a complex symbol (a chunk) of any size, this boosts the probability of sequences in which the parts of the complex symbol are contiguous. For example, if a message contains the binary complex symbol (*cats, black*), then a boost is applied to the probability of schemas such as *cats-black-three-these*, *black-cats-three-these*, *three-cats-black-these*… etc. If a message contains a ternary complex symbol, such as ((*cats, black*), *three*), the boost requires contiguity of both the inner binary complex, as well as contiguity of this binary with the third term. This would therefore apply a boost to sequences such as *cats-black-three-these*, *three-cats-black-these*, but would not apply to *cats-three-black-these*, where all elements of the complex symbol are contiguous, but the inner binary element (*cats, black*) is discontiguous.

The contiguity parameter is applied as a probabilistic bias, rather than a hard constraint, for two reasons. Firstly, varying the weight of the parameter helps us to understand its effect on sequencing outcomes. Secondly, while we assume that complex symbol contiguity is generally preferred, natural languages do sometimes allow discontiguous linearisation of signs that are presumably stored and compiled as holistic symbols. One clear example is English expletive infixation, as in *fan-fucking-tastic* (McCarthy, 1982); other examples are West Germanic lexicalised particle verbs, e.g. English *I picked the children up*, or German *Er schlägt das Wort im Wörterbuch nach,* 'He looks up (lit. hits to) the word in the dictionary'.[5] It seems that in human languages complex symbol contiguity is preferred, but violable.

The contiguity parameter is applied as a multiplier effect on the probabilities of possible linear outputs. As mentioned above, the model begins with all posssible sequences as equiprobable, which is implemented as a pool of candidate grammatical sequences that initially contains one token of every possible sequence, i.e. all 24 permutations Dem-Adj-Num-N, Dem-Num-Adj-N, Dem-N-Adj-Num… Linearisation of the current message works by making a random selection from the pool, and using the grammatical category sequence to linearise the symbols, e.g. Dem-N-Adj-Num → *these-cats-black-three.*

As we will see below, the consistency parameter will gradually change the contents of the category ordering pool. Meanwhile, the contiguity parameter acts as a temporary multiplier. For example, in the pool's initial state all grammatical sequences have probability 1/24 = 0.04. If the symbols to be linearised are {(*cats, black*), *three, these*}, and the contiguity multiplier is 5, the sequence will be drawn from a temporarily

[5] We are grateful to a reviewer for pointing out the relevance of particle verbs, and to Christian Döhler for the German example.

modified version of the pool, with 5 tokens of each sequence that satisfies {N, Adj} adjacency (e.g. Adj-N-Dem-Num, Dem-Adj-N-Num, etc). There are 12 such sequences, and this gives a probability of 5/72 = 0.07 to each of the preferred sequences, and 1/72 = 0.01 to each of the dispreferred sequences. This contiguity boosting applies only to the current message, with its particular chunking properties, rather than leaving a lasting impression on the pool of candidate sequences.

The consistency parameter, which prefers repeated use of the same grammatical sequences, incrementally changes the candidate pool as the encoder works its way through the ~500 messages. Once a linear sequence has been selected for each message (via the contiguity parameter, as described above), this selection has a permanent effect on the candidate pool. For example, if we are at the initial state with a single token for each sequence, and the first message is encoded as *these-cats-black-three*, the sequence Dem-N-Adj-Num will be boosted in the candidate pool for subsequent messages. With a consistency multiplier of 10, this sequence will now have probability 10/33 = 0.30, while all other sequences will have their probabilities accordingly reduced to 1/33 = 0.03. This is now the candidate pool for encoding the next message, which will again apply the contiguity parameter to modulate the probabilities according to the specific chunking of the next message.

Notice that the contiguity parameter is an effect on individual messages, based on their specific symbolic PMIs, whereas the consistency parameter is more systematic, leading gradually to generalised effects on all messages. The interaction of these parameters captures some of the way that natural languages generally favour consistent sequencing by grammatical category, but at the same time do show some item-specific ordering (Simpson & Withgott, 1986; Mansfield et al., 2020, 2022). Also note that the effect of the multipliers is quite large at the initial stages of the encoder, when the candidate pool contains few tokens, but becomes smaller as the system encodes more messages and the overall size of the pool increases. If the consistency parameter is strong enough, the model gradually converges on one consistent grammatical sequence, as probability changes gradually become insufficient to effect candidate selection.

4.4. Convergence on the same sequences as natural languages

The effect of the two linearisation parameters in concert is that complex symbols tend to be contiguous in transmission sequences, and the grammatical categories that are most often chunked into complex symbols will tend to be adjacent not just in those messages where the chunking applies, but also in other messages (cf. Culbertson et al., 2020, p. 710). However the two principles can be in competition if say, categories {N, Adj} overall tend to have higher PMI on average, but in a specific message, say {N, Num} has a higher PMI. This will be illustrated below.

The main test of our model is whether it tends to produce the same NP-internal orderings that are observed in natural languages (Dryer, 2018), and how the two linearisation parameters affect the outcome. Inspection of the encoder's output reveals that it does indeed produce typologically preferred grammatical category sequences, at a broad range of parameter settings. For example, with contiguity multiplier = 20 and consistency multiplier = 5, running 20 iterations of the encoder converged on typologically preferred sequences 17 times (for details on 'convergence', see below), and on typologically dispreferred sequences just 3 times. This shows that for these parameter weights, the encoder has the very attractive properties of producing typologically preferred sequences most of the time, but also being capable of producing typologically dispreferred sequences, just as these do occur occasionally in natural languages.

(2) *Sample of 20 converged sequences (* = typologically dispreferred)*

Dem-Adj-N-Num
Num-N-Adj-Dem
Dem-Adj-N-Num
Dem-Adj-N-Num
Num-N-Adj-Dem
Adj-N-Num-Dem
Adj-N-Num-Dem
Adj-N-Num-Dem
Num-N-Adj-Dem
Num-Adj-N-Dem

*Dem-N-Num-Adj
Num-N-Adj-Dem
Adj-N-Num-Dem
Adj-N-Num-Dem
N-Adj-Num-Dem
N-Adj-Num-Dem
*Dem-N-Adj-Num
Adj-N-Num-Dem
Num-N-Adj-Dem
*N-Adj-Dem-Num

The typologically preferred sequences represent a clear probabilistic bias in natural languages, rather than a hard constraint (cf. Bickel, 2015). As noted above, in Dryer's (2018) sample of natural languages 83% exhibited one of the typologically preferred sequences. To further test whether the encoder produces a similar probabilistic bias, it was iterated 20 times each over a range of parameter weights for the contiguity and consistency parameters, with the outcome of interest being what proportion of the iterations converged on a typologically preferred linear schema. The chunking threshold was also tested at seven different levels, but this did not significantly affect the results and therefore is not reported below.[6] The results illustrated here group over seven chunking threshold values, each iterated 20 times giving 140 iterations at each pair of contiguity and consistency parameter values.

Figure 3 illustrates the proportion of typologically preferred sequences output at each pairing of contiguity and consistency parameters.[7] The figure shows that there is a parameter range in which the encoder produces a similar probabilistic bias to that observed in natural languages. In particular, the encoder produces iterations typologically common orders in 91% of its iterations (percentages for specific parameter combinations range from 70–100, σ=7), when the consistency multiplier is in the range

[6] The chunking threshold described in section 4.2 was tested across the range of values that resulted in a moderate degree of chunking (average symbol count > 2). Different chunking threshold values within the range used made no noticeable difference to the results.

[7] We first tested parameter values heuristically to identify the range in which results were sensitive to parameter changes, namely contiguity multipliers in the range of 1–100, and consistent sequencing multipliers in the range of 1–20.

of 2–10, and the contiguity multiplier is in the range of 5–100. This is similar to the 83% of natural languages that exhibit these orderings. We call this parameter range the ‘sweet region’, indicated in Figure 3 by the dotted rectangle.

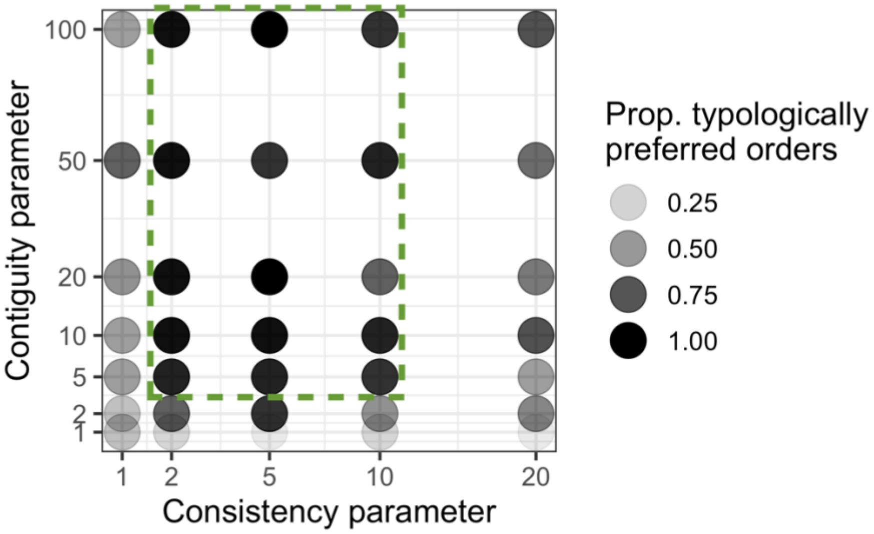

Figure 3. Sequencing of {N, Adj, Num, Dem}, as manipulated by the contiguity parameter and the consistency parameter. Darker shading indicates a higher proportion of iterations converging on one of the eight typologically common category sequences identified by Dryer (2018). The ‘sweet region’ is indicated by the dotted rectangle.

Figure 3 shows that the encoder fails to converge frequently on typologically common orders if the contiguity parameter is too low, indicating that the complex symbol contiguity principle is a requirement for typologically preferred sequencing in this model. The consistency parameter must have a weight above 1 (since multiplication by 1 has no actual effect), as otherwise the encoder tends not to converge on any particular category sequence at all. The encoder also struggles to converge on typologically preferred sequences if the consistency parameter is *too high* (> 10). This is because the encoder becomes over-eager to converge on a category sequence, and if it happens to encounter messages that have uncharacteristic chunking properties early in the message repertoire, it may converge on these sequences before more characteristically chunked messages have a chance to take effect.

4.5. The lack of harmonic dependencies

As outlined above, within the sweet region the encoder favours the same eight sequences as natural languages. However natural languages also exhibit further biases among these eight possibilities. In particular, Dryer’s (2018) sample shows that sequences are especially favoured where the noun is at one edge, and the other elements are sequenced in order of average PMI, i.e. N-Adj-Num-Dem or Dem-Num-Adj-N. Taking the N as the head and the others as dependents, we can say that these two particular sequences achieve a fully harmonic direction of dependencies.

The encoder does *not* favour the two fully harmonic sequences among the eight frequent sequences, but instead most favours Adj-N-Num-Dem and Num-N-Adj-Dem, accounting for 64% of convergence sequences in the sweet region. What these two sequences have in common is that the two categories that have highest average PMI with the noun, namely {Adj, Num}, are both adjacent to the noun. These sequences are presumably favoured by the encoder because they satisfy the complex symbol contiguity principle both for symbols such as (*cats, black*), and symbols such as (*wheels, four*).

This discrepancy between the encoder's output and natural languages suggests that our model is missing a crucial principle, namely harmonic dependency ordering (Greenberg, 1963; Culbertson & Newport, 2015; Jing et al., 2022). Harmonic ordering is satisfied by the sequences most favoured in natural languages, N-Adj-Num-Dem and Dem-Num-Adj-N, but not by the sequences most favoured by the encoder. We address this discrepancy in our follow-up study using natural language corpora.

4.6. Language-internal variation

The studies of NP ordering mentioned above focus on which ordering dominates in each language, while setting aside the question of language-internal variation. However studies that do report on variable ordering of NP elements suggest that this is quite common cross-linguistically (Rijkhoff, 2004). Even English, which is usually treated as having a fixed order Dem-Num-Adj-N, in fact exhibits some variation:

(3) a. The battery support and the four impressive wheels[8]
b. You already get some quite impressive four wheels for that[9]

We might suppose that examples such as (3b), which breaks the supposed rules of the English NP template, arise due to the complex symbol contiguity principle being triggered by (*wheels, four*), while examples such as (3a) are produced by the dominance of the consistent sequencing principle.[10]

As mentioned above, the encoder's linearisation algorithm is probabilistic, and has an initial state where all sequences are equally probable. Language-internal variation is therefore inherent to the model. However the consistency parameter tends to gradually bestow dominance on particular category sequences, or indeed on just one sequence. In the results above, the encoder is said to 'converge upon' particular category sequences, and we evaluate this based on whichever sequence is most frequent in the last fifty messages of the repertoire. Thus there are various degrees of convergence, as illustrated in Figure 4. This figure shows the same range or linearisation parameter values as above, with 140 iterations at each pair of values; but here for each the dot shading represents the

[8] https://affordablemedicalusa.com/blog/4-wheel-compact-travel-scooter-ensures-an-easy-driving-experience-with-adjustable-setup/

[9] https://flyctory.com/2021/06/29/driving-the-ferrari-488-with-pushstart-in-maranello/

[10] Another potential influence in this example is the elaboration of the adjective into a heavier element, *quite impressive*. We thank James Gray for pointing this out.

average degree of convergence in the iterations.[11] As we might expect, increases in the consistency parameter lead to greater convergence; but there is also a mild effect from the contiguity parameter, which slightly reduces convergence at its highest values (though this is difficult to distinguish in the figure shading). This is presumably because a strong contiguity parameter favours more message-specific sequencing, e.g. *cats-black-three-these* with N-Adj contiguity versus *wheels-four-white-those* with N-Num contiguity. The dashed rectangle in the lower-right corner indicates parameter values at which mean convergence reaches 100%, i.e. completely fixed category ordering in the last 50 messages. We call this the 'fixation region'. However we do not assume the fixation region to be an important outcome for validation of the model – as mentioned above, we do not know to what extent NP-internal ordering is rigid in natural languages.

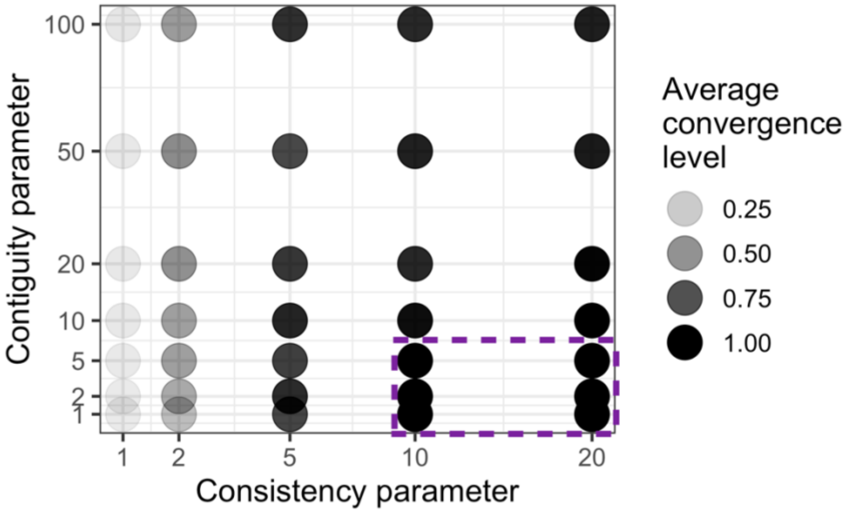

Figure 4. Degrees of convergence in categorical ordering, as manipulated by the contiguity parameter and the consistency parameter. Darker shading indicates a higher degree of convergence, i.e. the degree to which one particular category sequence dominates in the last fifty messages of each iteration of the encoder.

In this section we have reflected on free variation, but the principles in our model might also produce outcomes where specific lexical combinations have their own fixed ordering, which are not consistent by grammatical category (e.g. *black-cats*, *dogs-brown*). For this to occur, the structural priming mechanism would need to take into account specific lexical items, either instead of, or as well as, grammatical category labels (Mansfield & Kemp, in prep.) We therefore expect that our principles of linear ordering are quite compatible with the kinds of idiosyncratic templates described by Simpson (see §2 above), and can do so based on general principles rather than stipulative exceptions.

[11] The figure again collapses weightings of the chunking threshold, though in this case there did appear to be a very small effect, with more chunking slightly reducing the degree of convergence.

4.7. The model fails without informational chunking

To validate the claim that principles I–III can explain the sequencing of grammatical categories, we should be able to show that typologically preferred orders *fail* to emerge when any of the principles is absent from the model. This has already been done for the linearisation principles (II and III) in the previous section, where Figure 3 illustrated that if the parameter weight for either contiguity or consistency is too small, then the encoder fails to converge on typologically preferred category sequences.

We can additionally confirm that informational chunking is required to generate typologically preferred category sequences. Figure 5 illustrates the ouput of the encoder when it is run with the same overall degree of chunking, but with purely random as opposed to PMI-driven chunking. The figure shows no evidence of a bias towards typologically preferred sequences, the proportion of which mostly hovers around 30–40%, which is what we might expect from a random output, given that these orders constitute 33% (8/24) of possible orders. Furthermore the results appear to lose structural regularity with respect to the two linearisation parameters.

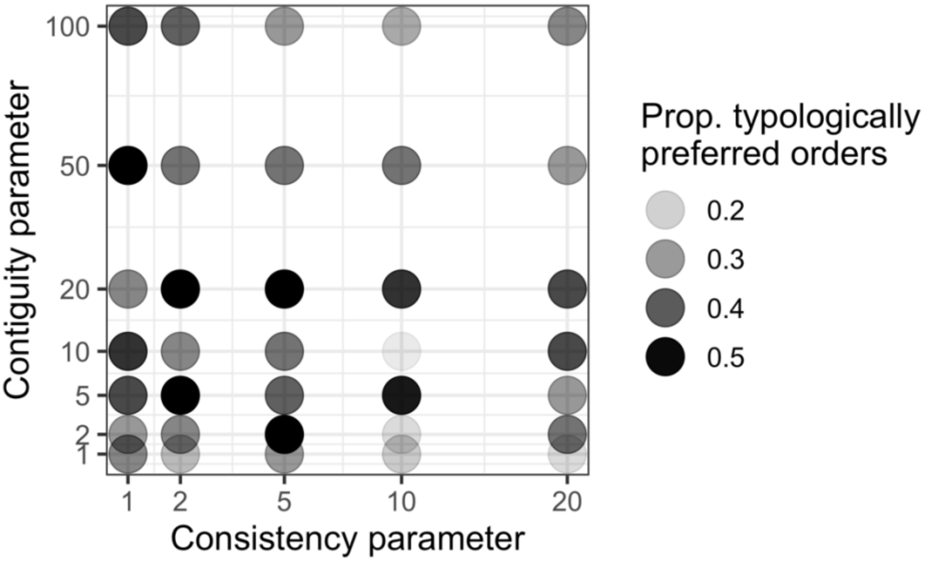

Figure 5. Output sequences when the encoder is run with random chunking, as opposed to PMI-driven chunking.

5. What have we done?

The encoder illustrated in this paper suggests that the NP-internal grammatical category orderings for {N, Adj, Num, Dem} can be explained as an emergent effect of independently evidenced principles of linguistic cognition. Most importantly, the encoder is not designed around any specific grammatical categories, and does not involve any ‘hard-coding’ of specific grammatical orderings. Rather, the orderings emerge from the inter-predictability of grammatically categorised symbols in the input data. We used artificial input data to capture the relevant inter-predictability distribution, which corpus research has shown that to be relatively constant in every one of 24 languages studied (Culbertson et al., 2020). Like Culbertson and colleagues, we assume that these symbolic inter-predictability patterns arise from semantic properties of adjectives, numbers and demonstratives, and how these reflect human experiences of the world. For example, adjectival qualities are more predictably associated with entities, while numerosities are less predictably associated with entities.

The principles implemented in the encoder are not specific to NP-internal grammar. Rather, they are intended to apply to any grammatical structure in which linear proximity corresponds to degrees of inter-predictability. As recent studies have revealed (see §3), this property of 'information locality' appears in a wide range of structures, and therefore the encoder modelled here has broad relevance to grammatical theory.

By formulating an explicit model of how inter-predictability can produce grammatical structure, this study advances the cause of usage-based or 'emergent' grammar. Much work in this tradition has discussed 'entrenchment' of linguistic structures from repeated use (e.g. Bybee, 2006; Diessel, 2019; Schmid, 2016). However implementation of formal models has been scarce in this tradition (though see Bod, 1998), and much of the literature focuses on specific lexical constructions rather than general grammatical patterns. The current study advances the field by addressing how specific messages interact with generalised grammatical schemas, and by testing an explicit model of such interaction.

The encoder illustrated above is also relevant to theoretical debates about hierarchical structure versus linear sequences. Simpson & Withgott (1986) proposed a role for linear schemas (templates) in grammar. The encoder formalises this proposal, and expands it beyond idiosyncratic morphological constructions. We have shown that our encoder is capable of converging upon consistent grammatical category sequences, while also noting that variation is inherent to the model. We also briefly observed that a similar encoder might produce lexically idiosyncratic sequences, using a modified version of the structural priming algorithm.

To reiterate, we do not claim that there is no hierarchical structure at all in syntax: unbounded recursive structures such as relative clauses, and reiterated complex units like the NP itself, both demand a hierarchical structure in which sets of signs are grouped together. But the assumption that all linguistic symbols fit into an underlying binary hierarchy seems unwarranted, and requires more complex structure than necessary whenever a linear schema will do just as well (Culicover & Jackendoff, 2005, p. 111). Our encoder does in fact produce hierarchical structures, but these are not 'syntactic' in any conventional sense. Instead the encoder applies recursive grouping to symbols in a storage-and-retrieval system, similar to the concept of a 'mental lexicon'. The hierarchical structures involve specific symbols, rather than generalised grammatical categories. Grammatical categories only play a role in the linearisation process, where structural priming favours the re-use of previous category sequences. Therefore generalised grammatical category schemas, i.e. morphosyntactic structure, is produced as the aggregate outcome of specific symbol combinations. It emerges iteratively and is never more than a distributional property of a probabilistic system.

The question of morphology versus syntax has not been addressed in this paper, though it has hovered in our peripheral vision. Simpson's templates were intended to be morphological, and have subsequently been taken up primarily within morphology (Nordlinger, 2010a; Stump, 1997), though see also (Good, 2016). Our suggestion is that principles of linear ordering do *not* respect the purported morphology/syntax distinction – if indeed such a distinction can be rigorously maintained (Haspelmath, 2011; Tallman, 2020). Future development of the encoder could attempt to capture the grammatical properties that are usually referenced to distinguish morphology from syntax. For example, it has been shown that selectional restrictiveness, one of the core properties involved in 'bound morphology', is associated with inter-predictability (Mansfield, 2021). This would contribute to a wider project that uses information theory to investigate the cluster of properties that tend to distinguish morphology from syntax (Ackerman & Malouf, 2013; Blevins, 2016). While attempts to formulate morphology/syntax as a categorical binary have been inconclusive, information-theory may present a fresh approach based on gradient properties.

Finally – what's missing? Several limitations were noted above, for example the use of artificial rather than natural corpus data, and the unrealistic restriction that each message consists of exactly four symbolic types, {N, Adj, Num, Dem}. The encoder illustrated here is merely a proof-of-concept, and its promising results should be validated by further research. But there are other more fundamental ways in which the encoder does not accurately represent human language. The encoder models a 'closed system' that outputs a fixed message repertoire, with no interaction between agents, or generations of learners. It is like a lonely individual who starts out with a proto-language of symbols and evolves into grammatically structured utterances by speaking into the void. Of course natural languages have not evolved in that way, but rather through cultural evolutionary processes shaped by the exchange of messages between many individuals, each of whom goes through an acquisition process. Where our model has a very simple pool of remembered grammatical sequences, natural language instead evolves by the gradual accretion of group memory over generations. There is also the matter of where grammatical categories such as {N, Adj, Num, Dem} come from, how they should be identified and whether they are even theoretically defensible (Anward, 2000; Croft, 2001; Bisang, 2010; Kenesei, 2020). All these will be challenges for future related research.

References

Ackerman, F., & Malouf, R. (2013). Morphological organization: The low conditional entropy conjecture. *Language*, *89*(3), 429–464.

Alexiadou, A., Haegeman, L., & Stavrou, M. (2008). Noun phrase in the generative perspective. In *Noun Phrase in the Generative Perspective*. De Gruyter Mouton. https://doi.org/10.1515/9783110207491

Ambridge, B. (2020a). Abstractions made of exemplars or ‘You’re all right, and I’ve changed my mind’: Response to commentators. *First Language*, *40*(5–6), 640–659. https://doi.org/10.1177/0142723720949723

Ambridge, B. (2020b). Against stored abstractions: A radical exemplar model of language acquisition. *First Language*, *40*(5–6), 509–559. https://doi.org/10.1177/0142723719869731

Anward, J. (2000). A dynamic model of part-of-speech differentiation. In P. M. Vogel & B. Comrie (Eds.), *Approaches to the Typology of Word Classes* (pp. 3–46). De Gruyter Mouton. https://doi.org/10.1515/9783110806120.3

Arnon, I., & Snider, N. (2010). More than words: Frequency effects for multi-word phrases. *Journal of Memory and Language*, *62*(1), 67–82. https://doi.org/10.1016/j.jml.2009.09.005

Aylett, M., & Turk, A. (2006). Language redundancy predicts syllabic duration and the spectral characteristics of vocalic syllable nuclei. *The Journal of the Acoustical Society of America*, *119*(5 Pt 1), 3048–3058. https://doi.org/10.1121/1.2188331

Baayen, R. H., & Schreuder, R. (Eds.). (2003). *Morphological structure in language processing*. Mouton de Gruyter.

Biber, D., Conrad, S., & Reppen, R. (1998). *Corpus linguistics: Investigating language structure and use*. Cambridge University Press.

Bickel, B. (2015). Distributional typology: Statistical inquiries into the dynamics of linguistic diversity. In B. Heine & H. Narrog (Eds.), *The Oxford handbook of linguistic analysis* (2nd edition, pp. 901–923). Oxford University Press.

Bisang, W. (2010). Word classes. In J. J. Song (Ed.), *Oxford handbook of linguistic typology*. Oxford University Press.

Blevins, J. P. (2016). *Word and paradigm morphology*. Oxford University Press.

Blumenthal-Dramé, A., Glauche, V., Bormann, T., Weiller, C., Musso, M., & Kortmann, B. (2017). Frequency and chunking in derived words: A parametric fMRI study. *Journal of Cognitive Neuroscience*, *29*(7), 1162–1177. https://doi.org/10.1162/jocn_a_01120

Bod, R. (1998). *Beyond Grammar: An Experience-Based Theory of Language*. CSLI Publications.

Brinton, L. J., & Traugott, E. C. (2005). *Lexicalization and language change*. Cambridge University Press. https://doi.org/10.1017/CBO9780511615962

Bybee, J. L. (2001). *Phonology and language use*. Cambridge University Press.

Bybee, J. L. (2006). From usage to grammar: The mind’s response to repetition. *Language*, *82*, 711–733.

Chomsky, N., & Lasnik, H. (2015). The theory of principles and parameters. In J. Jacobs, A. von Stechow, W. Sternefeld, & T. Venneman (Eds.), *Synatx: An international handbook of contemporary research* (pp. 506–569). De Gruyter Mouton. https://doi.org/10.1515/9783110095869.1.9.496

Church, K. W., & Hanks, P. (1990). Word association norms, mutual information, and lexicography. *Computational Linguistics*, *16*(1), 22–29.

Cinque, G. (2005). Deriving Greenberg’s Universal 20 and Its Exceptions. *Linguistic Inquiry*, *36*(3), 315–332. https://doi.org/10.1162/0024389054396917

Cinque, G., & Rizzi, L. (2009). The cartography of syntactic structures. In B. Heine & H. Narrog (Eds.), *The Oxford handbook of linguistic analysis* (pp. 52–66). Oxford University Press.

Contreras Kallens, P., & Christiansen, M. H. (2022). Models of language and multiword expressions. *Frontiers in Artificial Intelligence*, *5*. https://www.frontiersin.org/articles/10.3389/frai.2022.781962

Cover, T. A., & Thomas, J. A. (2002). *Elements of information theory* (Second edition). Wiley.

Croft, W. (2001). *Radical construction grammar: Syntactic theory in typological perspective*. Oxford University Press.

Crysmann, B., & Bonami, O. (2016). Variable morphotactics in information-based morphology. *Journal of Linguistics*, *52*(2), 311–374.

Culbertson, J., & Newport, E. L. (2015). Harmonic biases in child learners: In support of language universals. *Cognition*, *139*, 71–82. https://doi.org/10.1016/j.cognition.2015.02.007

Culbertson, J., Schouwstra, M., & Kirby, S. (2020). From the world to word order: Deriving biases in noun phrase order from statistical properties of the world. *Language*, *96*(3), 696–717.

Culicover, P. W., & Jackendoff, R. (2005). *Simpler syntax*. Oxford University Press.

Diessel, H. (2019). *The grammar network. How linguistic structure is shaped by language use*. Cambridge University Press.

Dik, S. C. (1997). *The theory of functional grammar: Part 1, the structure of the clause*. Mouton de Gruyter.

Dryer, M. S. (2009). The branching direction theory of word order correlations revisited. In S. Scalise, E. Magni, & A. Bisetto (Eds.), *Universals of Language Today* (pp. 185–207). Springer Netherlands. https://doi.org/10.1007/978-1-4020-8825-4_10

Dryer, M. S. (2018). On the order of demonstrative, numeral, adjective, and noun. *Language*, *94*(4), 798–833. https://doi.org/10.1353/lan.2018.0054

Everaert, M. B. H., Huybregts, M. A. C., Chomsky, N., Berwick, R. C., & Bolhuis, J. J. (2015). Structures, not strings: Linguistics as part of the cognitive sciences. *Trends in Cognitive Sciences*, *19*(12), 729–743. https://doi.org/10.1016/j.tics.2015.09.008

Fukumura, K., & Zhang, S. (2023). The interplay between syntactic and non-syntactic structure in language production. *Journal of Memory and Language*, *128*, 104385. https://doi.org/10.1016/j.jml.2022.104385

Futrell, R. (2019). Information-theoretic locality properties of natural language. *Proceedings of the First Workshop on Quantitative Syntax (Quasy, SyntaxFest 2019)*, 2–15. https://doi.org/10.18653/v1/W19-7902

Futrell, R., Levy, R. P., & Gibson, E. (2020). Dependency locality as an explanatory principle for word order. *Language*, *96*(2), 371–412. https://doi.org/10.1353/lan.2020.0024

Gibson, E., Futrell, R., Piantadosi, S. T., Dautriche, I., Mahowald, K., Bergen, L., & Levy, R. (2019). How efficiency shapes human language. *Trends in Cognitive Sciences*, *23*(5), 389–407.

Good, J. (2016). *The linguistic typology of templates*. Cambridge University Press.

Greenberg, J. (1963). *Universals of language*. MIT Press.

Grünwald, P. D. (2007). *The Minimum Description Length Principle*. MIT Press.

Hahn, M., Degen, J., & Futrell, R. (2021). Modeling word and morpheme order in natural language as an efficient trade-off of memory and surprisal. *Psychological Review*, *128*(4), 726–756. https://doi.org/10.1037/rev0000269

Hahn, M., Degen, J., Goodman, N. D., Jurafsky, D., & Futrell, R. (2018). An Information-Theoretic Explanation of Adjective Ordering Preferences. *Proceedings of the 40th Annual Meeting of the Cognitive Science Society*, 1766–1771.

Hahn, M., Mathew, R., & Degen, J. (2022). Morpheme ordering across languages reflects optimization for memory efficiency. *Open Mind: Discoveries in Cognitive Science (Accepted).*

Hale, J. (2001). A Probabilistic Earley Parser as a Psycholinguistic Model. *Second Meeting of the North American Chapter of the Association for Computational Linguistics*. NAACL 2001. https://aclanthology.org/N01-1021

Hartsuiker, R. J., Kolk, H. H. J., & Huiskamp, P. (1999). Priming Word Order in Sentence Production. *The Quarterly Journal of Experimental Psychology Section A*, *52*(1), 129–147. https://doi.org/10.1080/713755798

Haspelmath, M. (2011). The indeterminacy of word segmentation and the nature of morphology and syntax. *Folia Linguistica*, *45*(1), 31–80.

Hawkins, J. A. (2004). *Efficiency and complexity in grammars*. Oxford University Press. https://doi.org/10.1093/acprof:oso/9780199252695.001.0001

Hay, J. (2002). From speech perception to morphology: Affix ordering revisited. *Language*, *78*(3), 527–555. https://doi.org/10.1353/lan.2002.0159

Hoey, M. (2005). *Lexical priming: A new theory of words and language*. Routledge.

Hopper, P. (1987). Emergent grammar. *Proceedings of the Thirteenth Annual Meeting of the Berkeley Linguistics Society*, *13*, 139–157.

Hopper, P., & Traugott, E. C. (2003). *Grammaticalization* (Second Edition). Cambridge University Press.

Jaeger, T. F. (2010). Redundancy and reduction: Speakers manage syntactic information density. *Cognitive Psychology*, *61*(1), 23–62. https://doi.org/10.1016/j.cogpsych.2010.02.002

Jaeger, T. F., & Buz, E. (2017). Signal Reduction and Linguistic Encoding. In *The Handbook of Psycholinguistics* (pp. 38–81). John Wiley & Sons, Ltd. https://doi.org/10.1002/9781118829516.ch3

Jing, Y., Blasi, D. E., & Bickel, B. (2022). Dependency-length minimization and its limits: A possible role for a probabilistic version of the final-over-final condition. *Language*, *98*(3), 397–418. https://doi.org/10.1353/lan.0.0267

Kenesei, I. (2020). Life without word classes: On a new approach to categorization. In A. Bárány, T. Biberauer, J. Douglas, & S. Vikner (Eds.), *Syntactic architecture and its consequences II: Between syntax and morphology* (pp. 67–80). Language Science Press.

Kulmizev, A., & Nivre, J. (2022). Schrödinger's tree—On syntax and neural language models. *Frontiers in Artificial Intelligence*, *5*. https://www.frontiersin.org/articles/10.3389/frai.2022.796788

Kuperman, V., Bertram, R., & Baayen, R. H. (2010). Processing trade-offs in the reading of Dutch derived words. *Journal of Memory and Language*, *62*(2), 83–97. https://doi.org/10.1016/j.jml.2009.10.001

Levshina, N. (2022). *Communicative Efficiency: Language Structure and Use*. Cambridge University Press.

Levy, R. (2008). Expectation-based syntactic comprehension. *Cognition*, *106*(3), 1126–1177. https://doi.org/10.1016/j.cognition.2007.05.006

Mansfield, J. B. (2021). The word as a unit of internal predictability. *Linguistics*, *59*(6), 1427–1472. https://doi.org/10.1515/ling-2020-0118

Mansfield, J. B., & Kemp, C. (in prep.). *The emergence of grammatical structure from inter-predictability*.

Mansfield, J. B., Saldana, C., Hurst, P., Nordlinger, R., Stoll, S., Bickel, B., & Perfors, A. (2022). Category clustering and morphological learning. *Cognitive Science*, *46*(2), e13107. https://doi.org/10.1111/cogs.13107

Mansfield, J. B., Stoll, S., & Bickel, B. (2020). Category clustering: A probabilistic bias in the morphology of argument marking. *Language*, *96*(2), 255–293.

Marslen-Wilson, W. D. (2007). Morphological processes in language comprehension. In *The Oxford handbook of psycholinguistics* (pp. 175–193). Oxford University Press.

McCarthy, J. J. (1982). Prosodic structure and expletive infixation. *Language*, *58*(3), 574–590.

Nordlinger, R. (2010a). Agreement in Murrinh-Patha serial verbs. In Y. Treis & R. De Busser (Eds.), *Selected Papers from the 2009 Conference of the Australian Linguistic Society*.

Nordlinger, R. (2010b). Verbal morphology in Murrinh-Patha: Evidence for templates. *Morphology*, *20*(2), 321–341.

Perlmutter, D. M. (1971). *Deep and surface structure constraints in syntax*. Holt, Rinehart & Winston.

Pijpops, D., Smet, I. D., & Velde, F. V. de. (2018). Constructional contamination in morphology and syntax: Four case studies. *Constructions and Frames*, *10*(2), 269–305. https://doi.org/10.1075/cf.00021.pij

Plag, I., & Baayen, H. (2009). Suffix ordering and morphological processing. *Language*, *85*(1), 109–152.

Rijkhoff, J. (2004). *The noun phrase*. Oxford University Press.

Schmid, H.-J. (2016). *Entrenchment and the psychology of language learning, How we reorganize and adapt linguistic knowledge*. De Gruyter Mouton. https://doi.org/10.1515/9783110341423

Seržant, I. A., & Moroz, G. (2022). Universal attractors in language evolution provide evidence for the kinds of efficiency pressures involved. *Humanities and Social Sciences Communications*, *9*(1), Article 1. https://doi.org/10.1057/s41599-022-01072-0

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, *27*(3), 379–423.

Simpson, J., & Withgott, M. (1986). Pronominal Clitic Clusters and Templates. In H. Borer (Ed.), *The syntax of pronominal clitics* (Vol. 19). Academic Press.

Sinclair, J. (1991). *Corpus, concordance, collocation*. Oxford University Press.

Stump, G. T. (1997). Template morphology and inflectional morphology. In G. Booij & J. van Marle (Eds.), *Yearbook of Morphology 1996* (pp. 217–241). Kluwer Academic Publishers.

Szmrecsanyi, B. (2006). *Morphosyntactic persistence in spoken English: A corpus study at the intersection of variationist sociolinguistics*. Mouton de Gruyter.

Tallman, A. J. R. (2020). Beyond grammatical and phonological words. *Language and Linguistics Compass*, *14*(2), e12364. https://doi.org/10.1111/lnc3.12364

ten Hacken, P. (2019). The mental lexicon in Jackendoff's parallel architecture. In P. ten Hacken (Ed.), *Word Formation in Parallel Architecture: The Case for a Separate Component* (pp. 3–22). Springer International Publishing. https://doi.org/10.1007/978-3-030-18009-6_2

Van Gompel, R. P. G., & Arai, M. (2018). Structural priming in bilinguals. *Bilingualism: Language and Cognition*, *21*(3), 448–455. https://doi.org/10.1017/S1366728917000542

Wells, R. S. (1947). Immediate constituents. *Language*, *23*(2), 81–117. https://doi.org/10.2307/410382

Wray, A. (2002). *Formulaic language and the lexicon*. Cambridge University Press.

Wray, A. (2017). Formulaic sequences as a regulatory mechanism for cognitive perturbations during the achievement of social goals. *Topics in Cognitive Science*, *9*(3), 569–587. https://doi.org/10.1111/tops.12257

Zipf, G. K. (1949). *Human behavior and the principle of least effort: An introduction to human ecology*. Addison-Wesley Press.