

Computational Complexity of Natural Languages: A Reasoned Overview

António Branco

University of Lisbon

NLX-Natural Language and Speech Group

Departamento de Informática, Faculdade de Ciências, Universidade de Lisboa

Campo Grande, 1749-016 Lisboa, Portugal

antonio.branco@di.fc.ul.pt

Abstract

There has been an upsurge of research interest in natural language complexity. As this interest will benefit from being informed by established contributions in this area, this paper presents a reasoned overview of central results concerning the computational complexity of natural language parsing. This overview also seeks to help to understand why, contrary to recent and widespread assumptions, it is by no means sufficient that an agent handles sequences of items under a pattern $a^n b^n$ or under a pattern $a^n b^m c^n d^m$ to ascertain ipso facto that this is the result of at least an underlying context-free grammar or an underlying context-sensitive grammar, respectively. In addition, it seeks to help to understand why it is also not sufficient that an agent handles sequences of items under a pattern $a^n b^n$ for it to be deemed as having a cognitive capacity of higher computational complexity.¹

1 Introduction

The complexity of natural language became a specific topic of scientific inquiry and progress when it was addressed from the perspective of its computational processing. The study of the computational complexity of natural language was pioneered by Noam Chomsky in the late 1950's and has advanced since then with a growing body of established results.

This paper aims to provide a concise overview of these results. Its immediate motivation is the ongoing upsurge of research interest on the complexity of natural language. Examples of this interest include an edited volume on *Measuring Linguistic Complexity* (Newmeyer and Preston, 2014) and a special journal issue on *Pattern Perception and Computational Complexity* (Fitch et al., 2012b), and the reviews therein on *Computational Complexity in the Brain* (Chesi and Moro, 2014), on *The Neurobiology of Syntax* (Petersson and Hagoort, 2012) and on *Artificial Grammar Learning Meets Formal Language Theory* (Fitch et al., 2012a), among others.

In the context of this renewed interest many studies appear to be misled by misunderstandings of relevant mathematical notions and proofs, thus inducing misinterpretations of empirically gathered evidence. A case in point is the wide-spreading assumption that it is sufficient that an agent handles sequences of items under a pattern $a^n b^n$ or under a pattern $a^n b^m c^n d^m$ in order to ascertain ipso facto that this is the result of at least, respectively, an underlying context-free grammar or an underlying context-sensitive grammar. Another important related case to note is the assumption, more or less explicit, that an agent can be shown to master cognitive skills of higher computational complexity if it is shown to be able to handle a few sequences that conform to the pattern $a^n b^n$.

This paper aims at providing a reasoned overview on the computational complexity of natural language parsing. As these results are dispersed within an array of publications, putting them together in an articulated presentation will allow these past advances to be beneficial to forthcoming research. In this regard, we also seek to reinforce the momentum around the topic of natural language complexity.

In the next Section 2, we report on how the intricacies of natural language processing have been circumscribed when it comes to address its computational complexity, and in Section 3, we present the criteria to ascertain different levels of computational complexity.

¹This paper was partly supported by the PORTULAN/CLARIN Infrastructure and by the ANI/3279/2016 grant.

The key evidence that supports the discussion around the level of computational complexity of natural language parsing is presented in Section 4, and how this evidence has received different interpretations and supported different research programs is discussed in Section 5. The paper closes with final remarks in Section 6.

The presented overview also has a dissemination purpose. In order to reach a broad audience, some formal details are left out. The references provided should allow interested readers to dive into the relevant details if they wish to explore them further.

2 Processing problems

Human language is an entity of the natural world and to know within which boundaries its computational complexity lies it is necessary to understand how its processing takes place, and vice-versa. There is various empirical evidence upon which to draw hypotheses about the processing of natural language. This ranges from latency times obtained in experimental settings from a population of subjects to individual linguistic judgments, and includes quantitative data collected from corpora or images and recordings of neurological activity in the brain, among others. In the current state of our scientific knowledge about natural language, the empirical data uncovered thus far have been accounted for by different hypotheses and research frameworks concerning the processing of natural language. To a certain extent, the cogency of the conclusions about natural language computational complexity are dependent on the corresponding framework-internal assumptions and primitives.

It is also worth noting that the processing of natural language is unlikely to constitute a single monolithic procedure. For instance, taking into account perception — which permits the mapping of a linguistic form into the linguistic meaning it conveys —, several procedures are likely to be involved and interacting among each other (e.g. the detection of the different phonemes, their grouping into individual lexemes, the grouping of lexemes into phrases, the compositional calculation of their meaning from the meaning of their parts, etc.) All such different dimensions and sub-problems of language processing do not have necessarily to be addressed by a single computational method or procedure, or by different solutions of the same level of computational complexity.

The chances of finding firm results on the complexity of language may thus be as much higher as the sub-procedure under consideration is simpler, and as the empirical evidence is more elementary and less controversial, i.e. less prone to possibly contingent framework-driven interpretation or accommodation.

Important results have been obtained when the issue of complexity is addressed by studying what is known as the recognition problem: given a string s of lexical forms of a natural language L , how complex is the procedure to determine whether or not s is a sentence of L ?

Addressing the computational complexity of natural language from this perspective has the methodological advantage that the empirical evidence needed for its investigation is quite unequivocal and framework-independent, as it requires taking into account just strings of lexemes forming sentences.

One should not lose sight though of this methodological option and of the possible scope of its contribution concerning the eventual understanding of the complexity of natural language. When put into perspective with respect to the vast intricacies of human language processing, recognizing a string of lexical forms as a sentence is certainly a simple sub-procedure. Other sub-procedures are expected to be called into play in the global processing of language. It is also worth noting that the overall level of complexity of human language processing is not lower than the level of complexity of its more complex sub-procedures, on the one hand, and on the other hand, it is possible that some of these procedures have higher complexity than the recognition procedure.

Thus, whatever results one may eventually arrive at when researching the complexity of the recognition problem, they should be taken as representing a lower bound of the overall computational complexity of natural language.

3 Complexity levels

For the sake of perspicuity, the recognition problem is rendered as a set membership problem. When for methodological purposes, the empirical evidence to be taken into account is confined to strings of

lexemes, a language L lends itself to be regarded as the set S_L whose elements are precisely those strings of lexemes that are its sentences. Seeking a computational solution for the problem whether a string of lexemes s is recognized as being a sentence of language L is thus seeking a solution for the decision whether the string s is a member of the set S_L .

This would be a problem with an immediate, even if not efficient, brute force solution in case a human language could be extensionally presented as a listing with all and only its member sentences: it would simply require exhaustively scanning that list seeking for the input string. But as there is no clear size boundary for the possible longest grammatical sentences, that is not practically viable and the set of sentences of a language has rather been presented under an intensional definition. Such a definition relies upon a number of empirically motivated regularities and criteria determining the conditions for membership, which form a finite set of properly defined rules. This set of rules constitutes a grammar for the language.

Accordingly, a solution for the membership problem turns out to consist of designing a parser that takes as input a string s and a grammar G_L for the language L and after a finite number of steps delivers the answer *yes* in case s belongs to the set S_L defined by G_L , and the answer *no* otherwise. Under this methodological setup, a first move in assessing the computational complexity of the processing of a language consists of determining the complexity of the least possible complex parser for a grammar of that language.²

In this connection, it has been common practice to use a threefold computational complexity hierarchy as proposed by (Chomsky, 1956) that groups grammars into regular, context-free and context-sensitive types. All regular grammars are context-free grammars, and the set of all languages defined by the former are properly included in the set of all languages defined by the latter. Similar considerations hold with respect to context-free and context-sensitive languages, respectively.

In general terms that fit the purpose of the current overview, while no practical parser (i.e. with so called tractable computational complexity) could be found for every context-sensitive grammar, the best parsers for any regular or context-free grammar are practical solutions for the membership problem, with the best parser for regular grammars being a comparatively very efficient one.

In particular, the most efficient parsing algorithm for context-free grammars has polynomial (cubic) complexity, while best parsers for regular grammars have linear complexity — with time for obtaining a solution for a problem instance of size n (i.e. sentences with n lexemes) being around a value proportional to n^3 and n , respectively, in the worst case (Grune and Jacobs, 2007; Nederhof and Satta, 2010; Pratt-Hartmann, 2010).

This complexity hierarchy has been a yardstick used to help determine the complexity of the solution for the recognition problem in natural language. Assessing the level of complexity for this solution turns out to thus consist of empirically clarifying what type of grammar is suited to cope with this problem.

4 Grammar types

The claim that natural languages are not strictly regular, i.e. that they are supra-regular, was put forward in (Chomsky, 1956), and empirical elements from the English language in support of it can also be found in (Gazdar and Pullum, 1987, p. 394), or in the more accessible textbook (Partee et al., 1993, p. 477). An argument can be presented as follows.

4.1 Supra-regular

Consider the following sequence of English example sentences built by successively embedding into each other direct object relative clauses modifying subjects:

The cat escaped.

The cat [the dog bit] escaped.

The cat [the dog [the elephant stepped over] bit] escaped.

²As possible starting points on this, see (Hopcroft et al., 2001; Sudkamp, 2006; Wintner, 2010). Some authors, like (Sampson and Barbaczy, 2014), stress the dynamic nature of grammars in individuals and that the set of sentences of a language may have flexible boundaries. Some parsing procedure is always in place though, that allows speakers to distinguish, for instance, between different dialects and variants of a given language.

The cat [the dog [the elephant [the mouse frightened] stepped over] bit] escaped.

...

Based on these examples, and letting

$A = \{the\ dog,\ the\ elephant,\ the\ mouse,\ the\ fly,\ \dots\}$

$B = \{bit,\ stepped\ over,\ frightened,\ chased,\ \dots\}$

be finite sets of simple noun phrases and transitive verbs, respectively, the following infinite subset of English can be defined

$E' = \{the\ cat\ a^n b^n\ escaped \mid n \geq 0\}$

where a^n and b^n are any finite sequences of size n of concatenated members of A and B . Notice that E' is the intersection of the set E , containing all sentences of English, with the following regular language

$R = \{the\ cat\ a^* b^* \ escaped\}$

where a^* and b^* are finite sequences of any size of concatenated members of A and B , respectively. Given that regular sets are closed under the operation of intersection, that E' results from the intersection between R and E , and that E' is not regular,³ hence set E , with English sentences, is not regular.

While it is not practically feasible to check this result for every one of the over 7 000 existing languages in the world (Lewis et al., 2015), it is worth noting that this argument has been easily replicated with other types of syntactic constructions besides the center-embedded relative clauses above, and also for natural languages other than English ((Gazdar and Pullum, 1987, p.395); (Partee et al., 1993, p.478)).

In this connection it is worth noting that (Fitch and Hauser, 2004), seconded by (Gentner et al., 2006), proposed that the divide between regular and supra-regular computational process is the key to tell the difference between non-human and human-like cognitive capacities. This claim was based on arguments of the sort just described.⁴

In the search for the possible place of natural languages in the Chomsky hierarchy of computational complexity, the above argument leads to the next compelling question, whether natural languages are not context-free, i.e. whether they are supra-context-free (besides being supra-regular).

4.2 Supra-context-free

For three more decades, different attempts were made to support the claim that natural languages are supra-context-free, resorting to data from English comparatives (Chomsky, 1963), Mohawk noun-stem incorporation (Postal, 1964), "respectively" constructions (Bar-Hillel and Shamir, 1964; Langendoen, 1977), Dutch embedded verb phrases (Huybregts, 1976; Huybregts, 1984; Bresnan et al., 1982), number Pi (Elster, 1978), English "such that" clauses (Higginbotham, 1984), or English sluicing clauses (Langendoen and Postal, 1985). Those that were to be eventually retained as the best arguments are based on reduplication in noun formation in Bambara (Culy, 1985), and on Swiss German embedded infinitival verb phrases (Shieber, 1985).⁵

The argument based on Swiss German data is as follows. Consider the following sequence of example sentences built by successively embedding verb phrases in subordinate clauses (-DAT and -ACC signal dative and accusative case, respectively):

³The proof that $a^n b^n$ is not regular resorts to the following Pumping Lemma for Regular Languages: Let L be a regular language. Then there exists a constant c (which depends on L) such that for every string w in L of length $l \geq c$, we can break w into three subsequences $w = xyz$, such that y is not an empty string, the length of xy is less than $c + 1$, and for all $k \geq 0$, the string $xy^k z$ is also in L (Hopcroft et al., 2001, p.126). Intuition for the proof: however the members of E' of length longer than c are broken, no subsequences of them can be found that consistently match a pattern $xy^k z$ (for a proof, see (Sipser, 2013, p.80)).

The intended proof that E' (and hence E , i.e. the English language) is not regular has its grip in case E' is considered to be infinite: see Section 5.2 below on the empirical grounds to eventually dispute this.

⁴The validity of the argument given the experimentally elicited data obtained to sustain it was strongly challenged however (Lieberman, 2004; Coleman et al., 2004; Pinker and Jackendoff, 2005). An overview can be found in (Fitch et al., 2012a).

⁵For extended overviews and critical assessment, see (Pullum and Gazdar, 1982; Pullum, 1984; Partee et al., 1993).

Jan säit das mer em Hans es huus haend wele hälfe aasrüiche.

Jan said that we the Hans-DAT the house-ACC have wanted help paint

Jan said that we have wanted to help Hans paint the house.

Jan säit das mer d'chind em Hans es huus haend wele laa hälfe aasrüiche.

Jan said that we the children-ACC the Hans-DAT the house-ACC have wanted let help paint

Jan said that we have wanted to let the children help Hans paint the house.

...

Based on these examples, and letting

A = {*d'chind*, ...}

B = {*em Hans*, ...}

C = {*laa*, ...}

D = {*hälfe*, ...}

be finite sets of accusative noun phrases (A), dative noun phrases (B), accusative object taking transitive verbs (C), dative object taking transitive verbs (D), respectively, the following subset of Swiss German can be defined :

$$G' = \{Jan\ säit\ das\ mer\ a^n b^m\ es\ huus\ haend\ wele\ c^n d^m\ aasrüiche \mid n, m \geq 0\}$$

Notice that G' is the intersection of the set G , with all sentences of Swiss German, with the following regular language R

$$R = \{Jan\ säit\ das\ mer\ a^* b^* es\ huus\ haend\ wele\ c^* d^* aasrüiche\}$$

Given that context-free sets are closed under intersection with regular sets, that G' results from the intersection between R and G , and that G' is not context-free, hence the set G , with Swiss German sentences, is not context-free.⁶

5 Research programs

For the purpose of gaining insight into the computational complexity of natural language processing, the inquiry reported above focused on the complexity of recognizing a string of lexemes as a sentence. Its outcome turns out to be methodologically productive as it helps to uncover what appear as interesting constraints concerning the nature and processing of natural languages. The way these constraints have been addressed and accounted for has been a key factor on how different types of grammatical research frameworks for natural language have been shaped.

5.1 Matching the complexity of the recognition problem

One possible research path has been to study and design natural language grammars that match the claim of supra-context-freeness with as low a cost as possible in terms of computational complexity. This implies going slightly beyond context-freeness, just to the extent needed for the recognition problem of all sentences to receive a solution.

This goal has been pursued by exploring the fact that not all context-sensitive languages beyond context-freeness require a grammar whose parser is of non practical complexity.⁷ Grammar formalisms of this type have then been used to develop computational grammars for natural languages able to handle known grammar constructions beyond the power of context-free grammars, thus providing a constructive

⁶The proof that $a^n b^m c^n d^m$ is not context-free resorts to the following Pumping Lemma for Context-free Languages: Let L be a context-free language. Then there exists a constant c (which depends on L) such that if z is any string in L such that its length is at least c , then we can write $z=uvwx^i y$, subject to the following conditions: (i) the length of vw is at most c ; (ii) vx is not an empty string; (iii) for all $i > 0$, $uv^i wx^i y$ is in L (Hopcroft et al., 2001, p.275). Intuition for the proof: however the members of G' of length longer than c are broken, no subsequences of them can be found that consistently match the pattern $uv^i wx^i y$. (for a proof, see (Sipser, 2013, p.128)).

The intended proof that G' (and hence G) is not context-free has its full grip in case G' is considered to be an infinite set: see Section 5.2 below on the empirical grounds to eventually dispute this.

⁷For a critical overview, see (Gazdar and Pullum, 1987; Partee et al., 1993, Chap. 21).

argument that such linguistic constructions do not necessarily push the processing of natural language to computationally unpractical solutions.

This is the line of research pursued most notably i.a. by the GPSG⁸ framework (Gazdar and Pullum, 1987), and by its successor, the HPSG⁹ framework (Pollard and Sag, 1987; Pollard and Sag, 1994).

5.2 Approximating the complexity of the recognition problem

Another research path is based on a different position with respect to the interpretation of the results presented in the previous section.

First, it is worth noting that the more solid empirical evidence interpreted as possibly pushing natural language complexity beyond context-freeness is the so-called *cross-serial dependencies* mentioned above, with respect to Swiss German. It took not only almost three decades of research effort to arrive at the results reported in (Culy, 1985; Shieber, 1985), as no other kinds of constructions were identified as having the same sort of implication in terms of complexity. Moreover, the cross-serial dependencies between verb phrases and their complements get harder, if not impossible, to be recognized by native speakers beyond triple embedding (Shieber, 1985, p.329).

These circumstances have been invoked to support the view that natural languages are in their essence within the context-free level of complexity: put colloquially, a language that has a *finite* subset of sentences matching the pattern $a^n b^m c^n d^m$ (thus with $0 \leq n, m \leq k$ for some constant k), and that otherwise (i.e. expunged from that subset) can be described by a context-free grammar even when including that subset — note that *there is no requirement that the language be finite, only that the number of embeddings is finite*.¹⁰

Second, the *center-embedding constructions* pushing natural language complexity beyond regular grammar, in turn, are easy to replicate in different languages with different kinds of constructions. Nevertheless, also here, human speakers find themselves at odds to recognize sentences with more than a few embeddings. A vast arrays of empirical research results are confluent in reinforcing this observation, showing “that sentences with more than two centre embeddings are read with the same intonation as a list of random words, cannot easily be memorized, are difficult to paraphrase and comprehend, and are sometimes paradoxically judged ungrammatical” (Petersson and Hagoort, 2012, p.1976).

In this respect, it is interesting to note the contrast between, on the one had, the increasing difficulty of processing sentences in the sequence of center embeddings, used to argue for the supra-context-freeness of natural languages

The cat escaped.

The cat [the dog bit] escaped.

The cat [the dog [the elephant stepped over] bit] escaped.

The cat [the dog [the elephant [the mouse frightened] stepped over] bit] escaped.

...

and, on the other hand, the much lower difficulty in processing a syntactically similar sequence but now with *peripheral right-embedding*¹¹

The cat escaped.

The cat [that bit the dog] escaped.

The cat [that bit the dog [that stepped over the elephant]] escaped.

The cat [that bit the dog [that stepped over the elephant [that frightened the mouse]]] escaped.

...

⁸Generalized Phrase Structure Grammar.

⁹Head-driven Phrase Structure Grammar.

¹⁰Intuition for the proof: (i) recall that by definition any grammar has *finite* sets of variables, terminals and rules, (ii) note that any string along a pattern of type a^n can be accounted for by n grammar rules of type $AN_i \rightarrow a AN_{i+1}$, with $1 \leq i \leq n$, and any sequence $a^n b^m c^n d^m$ can thus be accounted for with suitable successive application of the appropriate sets of rules of that kind, and (iii) recall that rules with the format $X \rightarrow a Y$ do not push grammars beyond the class of regular grammars (Sudkamp, 2006, p.196) and thus beyond the level of linear complexity in their application to the recognition problem.

¹¹For an overview of literature reporting on this differing cognitive effort, as evidenced by longer processing times, experienced by human speakers in handling these two patterns, see (Chesi and Moro, 2014, Section 3). As an aside yet interesting note, for the same given level of nesting, center embedding is empirically found in (Bach et al., 1986) to be even harder to process than cross-serial dependencies of the type uncovered in (Shieber, 1985).

This contrast has been used to support the view that there might be a finite upper bound also for center embedding in natural languages, in which case a regular grammar should be enough to describe these linguistic constructions.

Mutatis mutandis, the observation above applies here: a language that has a *finite* subset of sentences matching the pattern $a^n b^n$ (thus with $0 \leq n \leq k$ for some constant k), and that otherwise (i.e. expunged from that subset) can be described by a regular grammar, can be described by a regular grammar even when including that subset. Again, note that *there is no requirement that the language be finite, only that the number of embeddings is*.¹²

This view is further reinforced by the fact that peripheral embedding, though not center-embedding, can be accounted for by regular grammars (Langendoen, 1975; Gazdar and Pullum, 1987; van Noord, 1998).

These points, together with the observation that humans process language very efficiently in a time that approximates a linear function of the length of the sentences, support the claim that regular grammars can provide at least very good approximations to the description of natural languages. This is the line of research advocated in (Roche and Schabes, 1997; van Noord, 1998).

Although they are different, it is worth noting that this perspective and the one indicated in the previous subsection are not necessarily in conflict. The complementarity nature of the two has actually been explored under the rationale that less complex solutions should be used as much as possible until the point where resorting to more complex solutions turns out to be unavoidable with respect to the eventual nature of the sub-problems to be solved. Regular methods have been applied to shallow linguistic processing, whose outcome feeds augmented context-free grammars in charge of deep linguistic processing, responsible for yielding fully-fledged grammatical representations (Crysmann et al., 2002).

Nevertheless, when it comes to the accommodation of the results presented in the previous section, the largest divide is perhaps not so much between these two research programs as it is between them and a third, to be presented in the next subsection below.

5.3 The complexity of the recognition problem in a trade off

The two approaches described in the two subsections above result from different perspectives on empirical data supporting arguments on the complexity level. A third line of research calls instead for putting into perspective the complexity metric used. In particular, it is noted that the distinction between polynomial and exponential is a coarse-grained measure of complexity, that is based on an asymptotic notation and abstracts away from many varying details of the basic operations of different computing devices. As repeatedly warned in textbooks on computational complexity, this distinction is known to be a reliable indicator of the actual superior efficiency of algorithms for problem instances that are larger than a sufficiently large size, such that a polynomial growth of the time needed to complete its operation will never be outperformed in terms of efficiency by an exponential growth.¹³

In the case of sentence recognition, the size of a problem instance is determined by the number of words in the input candidate sentence. And when it comes to natural languages, the actual input problem instances are made of at most a few dozen of words each on average.

Under such circumstances, for the actual time required to find a solution to a recognition problem instance of this size, it is likely that it is the natural language grammar — with its considerable memory size requirements in terms of the number of rules to be accessed, the internal data structure to encode them, etc. —, rather than the parser, that turns out to be responsible for the largest share. Moreover, moving from weaker and more efficient (e.g. regular) to more powerful and less efficient (e.g. context-sensitive) grammar types permits that a given language may be described more succinctly by its grammar. Consequently, grammars well beyond context-freeness — even if requiring companion, exponential parsers — may process natural language *sentences of actual average size* faster than infra-context-sensitive ones.

Thus, given the comparatively very small size of the actual input to the recognition problem in natural languages (the average size of sentences), the key issue for matching the observed human parsing effi-

¹²The intuition for the proof is as in footnote 10.

¹³As possible starting points on this, see among many others ((Gutttag, 2013, Chap. 9) ; (Cormen et al., 2009, Chap. 3)) .

ciency is not finding the most efficient parsing algorithm to cope with the empirically observed data like those illustrated in the section above. Alternatively, it is finding the best trade-off between the level of complexity brought into the *overall sentence processing procedure* by the parsing algorithm, on the one hand, and on the other hand by other factors relevant given the small size of the input problems at stake — namely by the size and shape of the grammar. Accordingly, natural language grammar is very likely to be of a context-sensitive type, with its companion parser of exponential complexity.

This position is fully articulated in (Berwick and Weinberg, 1982).¹⁴ The LFG¹⁵ framework (Kaplan and Bresnan, 1982) is a research program that lends itself to be classified as a grammar framework admitting context-sensitive grammars for natural languages (Bresnan et al., 1982; Berwick, 1982).¹⁶

6 Final remarks

The programs of research on natural language grammar described above adopt different ways to accommodate results from research on the computational complexity of the recognition problem. Given the Chomsky complexity hierarchy for computable solutions, they fill the whole spectrum of hypothesis ranging from the position that the grammars of natural languages are regular to the positions that they are context-sensitive, also including the claim that they are basically context-free.

What these research programs and the argumentation supporting them bring to light is that, importantly, it is by no means sufficient that a linguistic construction instantiates, a language includes, or an agent handles sequences of items under a pattern $a^n b^n$ or under a pattern $a^n b^m c^n d^m$ to ascertain ipso facto that these patterns are the result or empirical evidence of at least, respectively, an underlying context-free grammar or an underlying context-sensitive grammar. Likewise, by themselves alone, they are not sufficient to ascertain cognitive skills of higher computational complexity.

To interpret the relevant empirical evidence here, *it is not only the shape of the patterns that matter; the possible length of the stretch made of iterated items and the size range of the input also matter.*

Of course, these observations also hold for artificial languages that happen to be mastered by humans and non-humans alike under experimental settings.¹⁷

Overlooking these results and research programs has misled many research efforts into a maze of misunderstandings of mathematical notions and proofs, and of concomitant misinterpretations of empirical data. This may very well be prolonged by the current revival of interest on the complexity of natural languages, with the programmatic insistence on pattern shape and with the continued overlooking of size and related efficiency issues (Fitch et al., 2012a; Chesi and Moro, 2014).

This should not, however, dispute that restricting the focus of inquiry to the recognition procedure has been a productive methodological move, one that has permitted new insights into the computational complexity of natural language. Yet, as noted at the outset, this is certainly just one of the possible sub-procedures involved in the wider task of natural language processing, helping to advance research on the lower bound of natural language complexity.

As *empirical data from more and, above all, better articulated sources of evidence* become available (e.g. contrasts in grammatical judgments, linguistic performance and behavioral scores, records of brain activity, neurological findings, etc.), one should expect that the number of working hypotheses about the computational complexity of natural language could be narrowed down provided that they are obtained in experimentation correctly informed by the underpinnings of parsing methodology and of the theory of computation.

¹⁴Though inspired by other kind of empirical evidence, in the overview in (Pettersson and Hagoort, 2012, p.1976), this is also what seems to be hinted at as an admissible hypothesis: "There are often interesting complex trade-offs between processing time and memory use in computational tasks, and understanding these might be of importance to neurobiology".

¹⁵Lexical Functional Grammar.

¹⁶The GB (Government and Binding) research framework and its successors in the scope of MP (Minimalist Program) (Chomsky, 1981; Chomsky, 1995) are deemed to embrace this position. These research traditions have been criticized though by not using a clearly defined grammar formalism, which could support the development of a computational grammar for which complexity issues can be determined (Johnson and Lappin, 1997; Johnson and Lappin, 1999; Lappin et al., 2000).

¹⁷For an overview on experimentation with artificial grammar learning, see ((Pettersson and Hagoort, 2012);(Fitch et al., 2012a, Sections 5 and 6)).

References

- Emmon Bach, Colin Brown, and William Marslen-Wilson. 1986. Crossed and nested dependencies in German and dutch: a psycholinguistic study. *Language and Cognitive Processes*, 1(4):249–262.
- Yehoshua Bar-Hillel and E. Shamir. 1964. Finite state languages: Formal representations and adequacy problems. In Yehoshua Bar-Hillel, editor, *Language and Information*, pages 87–98. Addison-Wesley.
- Robert Berwick and Amy Weinberg. 1982. Parsing efficiency, computational complexity, and the evaluation of grammatical theories. *Linguistic Inquiry*, 13:165–191.
- Robert Berwick. 1982. Computational complexity and lexical functional grammar. *American Journal of Computational Linguistics*, 8:97–109.
- J. Bresnan, R. Kaplan, S. Peters, and A. Zaenen. 1982. Cross-serial dependencies in dutch. *Linguistic Inquiry*, 13.
- C. Chesi and A. Moro. 2014. Measuring linguistic complexity. In Newmeyer and Preston (eds.), Chap. 13.
- Noam Chomsky. 1956. Three models for the description of language. *IRE Transactions on Information Theory*.
- Noam Chomsky, 1963. *Formal Properties of Grammars*, pages 323–418. John Wiley And Sons, Inc. In R. Luce, R. Bush and E. Galanter (eds.), *Handbook of Mathematical Psychology*.
- Noam Chomsky. 1981. *Lectures on Government and Binding*. Dordrecht, Foris.
- Noam Chomsky. 1995. *The Minimalist Program*. MIT Press, Cambridge.
- John Coleman, Greg Kochanski, Burton Rosner, and Esther Grabe. 2004. January. Letter to Science editor, <http://kochanski.org/gpk/papers/2004/FitchHauser/FitchHauserScienceLetter.pdf> Expanded in <http://kochanski.org/gpk/papers/2004/FitchHauser/>.
- Thomas Cormen, Charles Leiserson, and Ronald Rivest. 2009. *Introduction to Algorithms*. MIT Press, 3rd edition.
- B. Crysmann, A. Frank, K. Bernd, S. Mueller, G. Neumann, J. Piskorski, U. Schaefer, M. Siegel, H. Uszkoreit, F. Xu, M. Becker, and H. Krieger. 2002. An integrated architecture for shallow and deep processing. In *40th Annual Meeting of the Association for Computational Linguistics*, pages 441–448. ACL.
- Christopher Culy. 1985. The complexity of the vocabulary of bambara. *Linguistics and Philosophy*, 8:345–351.
- Jon Elster. 1978. *Logic and Society: Contradictions and Possible Worlds*. New York.
- W. Tecumseh Fitch and Marc Hauser. 2004. Computational constraints on syntactic processing in a nonhuman primate. *Science*, 303:377–380.
- W. Tecumseh Fitch, Angela Friederici, and Peter Hagoort. 2012a. Artificial grammar learning meets formal language theory: an overview. *Philosophical Transactions of the Royal Society*, 367:1933–1955.
- W. Tecumseh Fitch, Angela D. Friederici, and Peter Hagoort. 2012b. Pattern perception and computational complexity (special issue). *Philosophical Transactions of the Royal Society B*, 367.
- Gerald Gazdar and Geoffrey Pullum. 1987. Computationally relevant properties of natural languages and their grammars. *New Generation Computing*, pages 387–43.
- Timothy Gentner, Kimberly Fenn, Daniel Margoliash, and Howard C. Nusbaum. 2006. Recursive syntactic pattern learning by songbirds. *Nature*, 440:1204–1207.
- Dick Grune and Criel Jacobs. 2007. *Parsing Techniques: A Practical Guide*. Springer.
- John Guttag. 2013. *Introduction to Computation and Programming Using Python*. The MIT Press.
- James Higginbotham. 1984. English is not a context-free language. *Linguistic Inquiry*, 15:225–234.
- J. Hopcroft, R. Motwani, and J. Ullman. 2001. *Introduction to Automata Theory, Languages, and Computation*.
- R. Huybregts. 1976. Overlapping dependencies in dutch. Number 1. pp.24-65.
- Riny Huybregts. 1984. The weak inadequacy of context-free phrase structure grammars. In Germen J. de Haan, Mieke Trommelen, and Wim Zonneveld, editors, *Van Periferie Naar Kern*, pages 81–99. Foris Publications.

- David Johnson and Shalom Lappin. 1997. A critique of the minimalist program. *Linguistics and Philosophy*, 20:273–333.
- David Johnson and Shalom Lappin. 1999. *Local Constraints vs Economy*. CSLI Publications.
- Ronald Kaplan and Joan Bresnan. 1982. Lexical-functional grammar: A formal system for grammatical representation. In Joan Bresnan, editor, *The Mental Representation of Grammatical Relations*, pages 173–281.
- Terence Langendoen and Paul Postal. 1985. English and the class of context-free languages. *Computational Linguistics*, 10:177–181.
- Terence Langendoen. 1975. Finite-state parsing of the phrase-structure languages and the status of readjustment rules in grammar. *Linguistic Inquiry*, 5:533–554.
- Terence Langendoen. 1977. On the inadequacy of type-2 and type-3 grammars for human languages. In P. Hopper, editor, *Studies in Descriptive and Historical Linguistics*. John Benjamins.
- Shalom Lappin, Robert Levine, and David Johnson. 2000. The structure of unscientific revolutions. *Natural Language and Linguistic Theory*, 18:665–771.
- M. Paul Lewis, Gary F. Simons, and Charles D. Fennig, editors. 2015. *Ethnologue, Languages of the World*. SIL International, 18th edition.
- Marc Liberman. 2004. *Humans context-free, monkeys finite-state? Apparently not*. Language Log.
- Mark Nederkhof and Giorgio Satta. 2010. Theory of parsing. pages 105–130. In A. Clark, C. Fox and S. Lappin (eds.), *The Handbook of Computational Linguistics and Natural Language Processing*, Chap. 4.
- Frederick J. Newmeyer and Laurel B. Preston, editors. 2014. *Measuring Linguistic Complexity*. OUP.
- Barbara Partee, Alice ter Meulen, and Robert Wall. 1993. *Mathematical Methods in Linguistics*. Kluwer.
- Karl Magnus Petersson and Peter Hagoort. 2012. The neurobiology of syntax: beyond string sets. *Philosophical Transactions of the Royal Society*, 367:1933–1955.
- Steven Pinker and Ray Jackendoff. 2005. The faculty of language: What’s special about it? *Cognition*, 95:201–36.
- Carl Pollard and Ivan Sag. 1987. Information-based syntax and semantics. *CSLI Publication*.
- Carl Pollard and Ivan Sag. 1994. *Head-Driven Phrase Structure Grammar*. The University of Chicago.
- Paul Postal. 1964. Limitations of phrase structure. In J. Fodor and J. Katz, editors, *The Structure of Language: Readings in the Philosophy*. Englewood Cliffs, Prentice-Hall.
- Ian Pratt-Hartmann. 2010. Computational complexity in natural language. pages 43–73. In A. Clark, C. Fox and S. Lappin (eds.), *The Handbook of Computational Linguistics and Natural Language Processing*, Chap. 2.
- Geoffrey Pullum and Gerald Gazdar. 1982. Natural languages and context-free languages. *Linguistics and Philosophy*, 4:471–504.
- Geoffrey Pullum. 1984. On two recent attempts to show that english is not a cfl. *Computational Linguistics*, 10:182–188.
- Emmanuel Roche and Yves Schabes. 1997. *Finite-State Language Processing*. The MIT Press.
- Geoffrey Sampson and Anna Barbaczy. 2014. *Grammar without Grammaticality: growth and limits of grammatical precision*. De Gruyter Mouton, Berlin.
- Stuart Shieber. 1985. Evidence against the context-freeness of natural language. *Linguistics and Philosophy*, 8:333–343.
- Michel Sipser. 2013. *Introduction to the Theory of Computation*. Cengage Learning, 3rd edition.
- Thomas A. Sudkamp. 2006. *Languages and Machines: An Introduction to the Theory of Computer Science*. Pearson, Boston.
- Gertjan van Noord. 1998. *Algorithms for Linguistic Processing*. Alfa-informatica, Groningen. <http://odur.let.rug.nl/vannoord/alp/proposal/pion.html>.
- Shuly Wintner. 2010. Formal language theory. pages 11–42. Wiley-Blackwell. In Alexander Clark, Chris Fox and Shalom Lappin (eds.), *The Handbook of Computational Linguistics and Natural Language Processing*, Chap. 4.