Abstract

This paper investigates linguistic performance, from the viewpoint that the information processing in cognitive systems should be designed in terms of constraints rather than procedures in order to deal with partiality of information. In this perspective, the same grammar, the same belief and the same processing architecture should underly both sentence comprehension and production. A basic model of sentence processing, for both comprehension and production, is derived along this line of reasoning. This model is demonstrated to account for diverse linguistic phenomena apparently unrelated to each other, lending empirical support to the constraint paradigm.

1 Introduction

All the cognitive agents, with limited capacity for information processing, face partiality of information: Information relevant to their activities is only partially accessible, and also the distribution pattern of the accessible information is too diverse to predict. In sentence comprehension, for example, the phonological or morphological information may or may not be partially missing due to some noise, the semantic information may or may not be abundant because of familiarity or ignorance on the topics, and so forth. Thus the information distribution is very diverse and rather orthogonal to the underlying information structure consisting of the modules of morphological, syntactic, pragmatic, and other constraints.

This diversity of information distribution gives rise to a very complex, non-modular flow of information in cognitive processes, as information flows from places possessing information to places lacking information. In order to deal with this complexity, a cognitive system must be designed to include two different logical layers:

1. Information represented in terms of constraints, by abstracting away information flow.
2. A general processing mechanism to convey information across constraints, from places possessing information to places lacking it.

Non-modular flow of information may be captured on the basis of modular design of cognitive architecture, only by separating the representation of underlying information (as (1)) and flow of information (as (2)) from each other.

Procedural approaches break down under partiality of information, because procedures stipulate, and hence restrict, information flow. If one be it human or nature, were to implement such diverse information flow by procedural programming, the entire system would quickly become too complex to keep track of, failing to maintain the modularity of the system. This is what has always happened, for example, in the development of natural language processing systems.

The rest of the paper exemplifies the efficacy of the constraint paradigm with regard to natural language. We will first discuss a general picture of language faculty immediately obtained from the constraint-based view, and then derive a model of sentence processing neutral between comprehension and production. This model will be shown to fit several linguistic phenomena. Due to the generality of the perspective, the phenomena discussed below encompass apparently unrelated aspects of natural language.

2 Language and Constraint

From the constraint-based perspective immediately follows a hypothesis that the same constraints (i.e., lexical, syntactic, semantic, pragmatic, and whatever), corresponding to (1), and the same processing architecture, corresponding to (2), should underly both sentence comprehension and production. Other authors have expressed less radical stances. For instance, Kay [11] adopts two different grammars for parsing and generation. Our hypothesis is also stronger than Shieber's [16]; Although he proposes to share not only one grammar but also one processing architecture between the two tasks, this 'common' architecture is, unlike ours, parameterized so as to adapt itself to parsing and generation in accordance with different parameter settings.
As a corollary of our strong uniformity hypothesis, we reject every approach postulating any procedure specific to sentence comprehension or production. For instance, we disagree upon the ways in which the Determinism Hypothesis (DH) [12] has been instantiated so far. DH permits to assume only one partial structure of a sentence at a time, and the approaches along this line [2, 3, 12, 14] has postulated, beyond necessity, specific ways of disambiguation for specific types of ambiguity in sentence comprehension and production.

Instead we view sentence processing as parallel computation. When a sentence is either comprehended or produced, several partial structures of it, we assume, are simultaneously hypothesized. The degree of parallelism should be limited to fall within the small capacity of the short-term memory (STM), so that we obtain the same sort of predictions as we do along the determinist account. For instance, the difficulty in comprehending garden path sentences like (3) may be attributed to the difficulty of keeping some structural hypotheses in STM.

(3) The chocolate cakes are coated with tastes sweet.

As discussed below, our approach quantitatively estimates the difficulty in processing embedded constructions like (4) also on the basis of the memory limitation.

(4) The cheese the rat the cat the dog chased caught bit was rotten.

Since DH does not account for such difficulty, incidentally, it seems superfluous to postulate DH. We consider DH just as approximation of severe memory limitation, and avoid any stipulation of such a hypothesis.

3 A Common Process Model

Among the partial structures hypothesized during comprehension or production of a sentence, we pay attention to the maximal structures; the structures such that there is no larger structures. Here we say one structure is larger than another when the former includes the latter. For example, [s [NP Tom] [VP sleeps]] is larger than [s [NP Tom] VP]. Sentence processing, whether comprehension or production, is regarded as parallel construction of several maximal structures. Thus sentence processing as a whole is characterized by specifying what a maximal structure is.

We assume the grammatical structure of a sentence to be a binary tree. Here we identify a word with its grammatical category, so that a local structure, such as [NP Tom], is regarded as one node rather than a partial tree consisting of two distinct nodes. It is just for expository simplification that we assume binary trees. Our account can be generalized straightforwardly to allow n-ary trees. Further, the essence of our discussion below is neutral between the constituency-based approaches and the dependency-based approaches. Here we employ a representation scheme of the former type, without committing ourselves to the constituency-based framework.

From the general speculation below, it follows that a maximal structure should be the left-hand half of (5).

![Diagram](5)

This maximal structure consists of the path form \( S \) to \( A \) and the part to the left of this path, except for \( B_{d-1} \) and the nodes between \( B_{d-1} \) and \( A_i \) (those on the slant dotted lines) for \( 1 \leq i \leq d+1 \); \( A_i \) and the nodes between \( A_i \) and \( B_i \) are included in the maximal structure. Here \( B_0 \) and \( A_{d+1} \) stand for \( S \) and \( A \), respectively. \( A_i \) is a leftmost descendant (not necessarily the left daughter) of \( B_{d-1} \) or they are identical for \( 1 \leq i \leq d+1 \). \( B_i \) is a rightmost descendant (not necessarily the right daughter) of \( A_i \) for \( 1 \leq i \leq d \). Thus our model is similar to left-corner parser [1], though our discussion is not restricted to parsing.

This characterization of a maximal structure is obtained as follows. First note that a maximal structure involves \( n \) words and \( n-1 \) nonterminal nodes, for some natural number \( n \); In the maximal structure in (5), the connected substructure containing \( A_i \) (1 \( \leq i \leq d \)) contains as many nonterminal nodes as words, so that the maximal structure also contains as many nonterminal nodes as words, except for word \( A \). Note further that the entire sentence structure, being a binary tree, also involves one less nonterminal nodes than words. Accordingly, postulating \( n-1 \) nonterminal nodes versus \( n \) words in a maximal structure amounts to postulating that the words and the nonterminal nodes are processed at approximately constant speed relative to each other.\(^{1}\) The number of words is a measure of lexical information, and the number of nonterminal nodes is a measure of syntactic and semantic information, among others. Hence if all the types of linguistic information (lexical, syntactic, semantic, etc.) are processed at approximately the same relative speed, then a maximal process should include nearly as many words as nonterminal nodes.

This premise is justified, because if different types of information were processed at different speeds, then

\(^{1}\)The rate of \( n \) words versus \( n-1 \) nonterminals does not precisely represent the constant relative speed, but the discrepancy here is least possible and thus acceptable enough as approximation.
there would arise imbalance of information distribution across the corresponding different domains of information. Such imbalance should involve information flow from the domains with higher density to the domains with lower density of information distribution, when, as in the case of language, those domains of information are tightly related with each other. That is, information flow eliminates such imbalance, resulting in approximately the same speed of processing across different but closely related domains of information.

Now that we have worked out how many nodes a maximal structure includes, what is left is which nodes it includes. Let us refer to A in (5) as the current active word and the path from the root node S to the current active word as the current active path. It is natural to consider that a maximal structure includes the nodes to the left of the current active path, because all the words they dominate have already been processed. Thus we come up with the above formulation of a maximal structure, if we notice that the nodes on the solid-line part (including Ai) of the current active path in (5) are adjacent to nodes to the left of the current active path, whereas the other nodes on the current active path (those on the dotted lines, including Bi) do not except for the mother of A, which will be processed at the next moment.

4 Immediate Processing

According to this model, any word should be immediately processed, particularly in parsing, in the sense that corresponding amount of syntactic and semantic structure is tailored with little delay. The intrasentential status of a word is hence identified as soon as it is encountered. This contrasts with the determinist accounts which assume lookahead to deal with local ambiguity.

Empirical evidences support our position. In Marslen-Wilson’s [13] experiment, for instance, the subjects were asked to listen to a tape-recorded utterance and to say aloud what they hear with the shortest possible delay. Some subjects performed this task with a lag of only about one syllable, and yet their error reflected both syntactic and semantic context. For example, one of such a subjects said He had heard that the Brigade ... upon listening to He had heard at the Brigade ... Such a phenomenon cannot be accounted for in terms of the determinist accounts with fixed parsing procedures. In our model, it is explained by just assuming that only the most active maximal structure tailored by the subject survives the experimental situation.

5 Transient Memory Load

By transient memory load (TML) we refer to the amount of linguistic information temporarily stored in STM. The measurements of TML during sentence processing proposed so far include the depth of center embedding (CE) [5] and that of self embedding (SE) [15]. A syntactic constituent α is center-embedded in another syntactic constituent β when β = γαδ for some non-null strings γ and δ. We further say that α is self-embedded in β when they are of the same sort of category, say NP.

However, neither CE nor SE can explain why (6) is much easier to understand than (7).

(6) Tom knows the story that a man who lived in Helsinki and his wife were poor but they were happy.

(7) Tom knows that the story on the fact that the rumor that Mary killed John was false is funny.

Note that these sentences are of about the same length; the former consists of 20 words and the latter 19 words. Almost all my informants (including both native and non-native speakers of English) reported that (6) is easier to understand than (7). Those who felt contrariwise ascribed the difficulty of (6) to the ambiguity concerning the overall structure of the complement clause after that.

The approach based on CE fails to account for this difference, because the maximum CE depth of (6) and that of (7) are both 3, as is shown below.

(8) [9 Tom knows the story that [1 a man [2 who [3 lived in Helsinki] and his wife] were poor but they were happy]

(9) [9 Tom knows that [1 the story on the fact that [2 the rumor that Mary [3 killed] John] was false] is funny]

The maximum SE depth cannot distinguish these sentences:

(10) Tom knows [NP0 the story that [NF, a man who lived in [NP1 Helsinki] and his wife] were poor but they were happy]

(11) Tom knows that [NP0 the story on the fact that [NF, the rumor that [NP1 Mary] killed John] was false] is funny.

Our model provides a TML measure which accounts for the contrast in question. In order to plug a maximal structure with the rest of the sentence in a grammatical manner, one must remember only the information contained in the categories on the border between the maximal structure and the remaining context; i.e., categories Ai, the mother of Bi (1 ≤ i ≤ d) and A in (5). Thus the value of d in (5) could serve as a TML measure. As is illustrated in (12) and (13), in fact, the maximum of d is 2 and 3 for (6) and (7), respectively, explaining why (6) is easier. In (12) and (13), enclosed in boxes are the nodes corresponding to Ai, Bi (1 ≤ i ≤ d) and A when d is the maximum; i.e., 2 in the former and 3 in the latter.
Tom knows the story that he and his wife were poor. They were happy after they lived in Helsinki.

Tom knows that the story on the fact that the rumor was false. Mary killed John.
6 Language Acquisition

The Dutch language exhibits a type of cross-serial dependency (CSD) in subordinate clauses:

(14) ...dat Wolf de kinderen Marie zag helpen zwemmen
     "...that Wolf the children help Marie swim"

Our theory predicts that children learning Dutch come to recognize the CSD constructions as having the following structure, which coincides with the structure figured out by Bresnan et al. [4] based on an analysis of adult language.

(15)

\[
\begin{array}{c}
S \\
NP_0 \quad VP \\
X_0 \quad Z_0 \\
NP_1 \quad V_0 \quad Z_{n-1} \\
NP_{m-1} \quad NP_m \quad V_{n-1} \quad V_n
\end{array}
\]

Here \(V_0\) is a finite verb and \(V_i\) is an infinite verb for \(1 \leq i \leq n\). \(V_i\) is a causative verb or a perception verb for \(1 \leq i < n\). \(NP_1\) is the subject of \(V_i\) for \(0 \leq i \leq n\), and \(NP_i\) is an object of \(V_i\) for \(n < i \leq m\) (\(m \geq m\)). Note that \(NP_1, \ldots, NP_m\) and \(V_0, \ldots, V_n\) constitute right-branching structures dominated by \(X_0\) and \(Z_0\), respectively.

Let us look at how a child regards a simple CSD construction (16) to be (17), which is an instance of (15) for \(m = n = 1\).

(16) ...dat Wolf Marie zag zwemmen
     "...that Wolf Marie see-PAST swim-INF"

(17) is thus obtained by setting \(V_0 = V_{1}\), \(Z_0 = Y_0\), and \(Z_1 = V_1\).

Note that this reasoning essentially relies on our formulation of a maximal process. If a bottom-up model were assumed instead, for instance, there would be no immediate reason to exclude a structure, say, as follows.

(20)

\[
\begin{array}{c}
S \\
NP_0 \quad VP \\
Wolf \quad \uparrow \quad V_1 \\
\uparrow \quad \downarrow \\
Marie \quad zag \quad \quad V_0 \quad zwemmen
\end{array}
\]

According to our model, the relevant part of the most active maximal structure would look like the following

\[153\]
The above discussion can be extended to cover more complex cases (where \( m > 1 \) in (15)) in a rather straightforward manner, as is discussed by Hasida [6]. The structure under \( X_0 \) is tailored as a natural extension of the way an ordinary subordinating clause is processed, then \( V_0 \) is inserted beneath VP, following the ordinary structure of a subordinate clause together with the semantic information about the situation described, and \( V_i \) attaches near to \( V_{i-1} \) for \( 1 \leq i < n \) due to the semantic information again. The structure under \( Z_0 \) must be right-branching so that \( V_0 \) be the head of VP.

Also by reference to the current model, Hasida [7] further gives an account of the unacceptability of some unbounded dependency constructions in English which is hard to explain in static terms of linguistics.

7 Concluding Remarks

We have begun with a general constraint-based perspective about the cognitive mechanism, and shown that a model of sentence processing derived thereof, neutral between comprehension and production, accounts for several linguistic phenomena seemingly unrelated to each other. It has thus been demonstrated that the speculation to derive the model has empirical supports, lending justification for the constraint paradigm. In particular, our theory has been shown to be more adequate than the determinist approach, which must postulate a procedural design of the human language faculty.

A computational formalization of our model will be possible in terms of constraint programming, as discussed by Hasida et al. [8, 9, 17]. Most of the time, a natural language processing system in terms of procedural programming has been designed to be a series of a syntactic analysis procedure, a semantic analysis procedure, a pragmatic analysis procedure, and so on, in order to reflect the modularity of the underlying constraints. However, such a design imposes a strong limitation on information flow, restricting the system's ability to a very narrow range of context. One naturally attempts to remedy this so as to, say, enable the syntactic analysis module to refer to semantic information, but this attempt must destroy the modularity of the entire design, ending up with a program too complicated to extend or even maintain. Constraint paradigm seems to be the only way out of this difficulty.

References

[1] Aho, A. V. and Ullman, U. D. (1972) *The Theory of Parsing, Translation and Compiling*, Prentice-Hall.

[2] Berwick, R. C. and Weinberg, A. (1984) *The Grammatical Basis of Linguistic Performance*, MIT Press.

[3] Berwick, R. (1985) *The Acquisition of Syntactic Knowledge*, MIT Press.

[4] Bresnan, J. Kaplan, R. M., Peters, S. and Zaenen, A. (1982) 'Cross-serial Dependencies in Dutch,' *Linguistic Inquiry*, Vol. 13, pp. 613-635.

[5] Church, K. W. (1980) *On Memory Limitations in Natural Language Processing*, MIT/LCS/TR-245, Laboratory for Computer Science, Massachusetts Institute of Technology.

[6] Hasida, K. (1985) *Bounded Parallelism: A Theory of Syntactic Performance*, doctoral dissertation, University of Tokyo.

[7] Hasida, K. (1988) 'A Cognitive Account of Unbounded Dependency,' in *Proceedings of COLING’88*, pp. 231-236.

[8] Hasida, K. (1989) *A Constraint-Based View of Language*, presented at the First Conference on Situation Theory and its Applications.

[9] Hasida, K. and Ishizaki, S. (1987) 'Dependency Propagation: A Unified Theory of Sentence Comprehension and Generation,' *Proceedings of IJCAI’87*, pp. 664-670.

[10] Kaplan, R. M. (1972) 'Augmented Transition Networks as Psychological Models of Sentence Comprehension,' *Artificial Intelligence*, Vol. 3, pp. 77-100.

[11] Kay, M. (1985) 'Parsing in Functional Unification Grammar,' in Dowty, D., Karttunen, L. and Zwicky, A. M. (eds.) *Natural Language Parsing: Psychological, Computational, and Theoretical Perspectives*, Cambridge University Press.

[12] Marcus, M. P. (1980) *A Theory of Syntactic Recognition for Natural Language*, MIT Press.

[13] Marslen-Wilson, W. D. (1975) 'Sentence Perception as an Interactive Parallel Process,' *Science*, Vol. 189, pp. 226-228.

[14] McDonald, D. (1980) *Natural Language Production as a Process of Decision Making under Constraint*, Doctoral Dissertation, Laboratory of Computer Science, Massachusetts Institute of Technology.

[15] Miller, G. A. and Chomsky, N. (1963) 'Finite Models of Language Users,' in Luce, R. D., Bush, R. R., and Galanter, E. *Handbook of Mathematical Psychology*, Vol. II, pp. 419-491, John Wiley and Sons.

[16] Shieber, S. M. (1988) 'A Uniform Architecture for Parsing and Generation,' in *Proceedings of COLING’88*, pp. 614-619.

[17] Tuda, H., Hasida, K., and Sirai, H. (1989) 'JPSG Parser on Constraint Logic Programming,' *Proceedings of the European Chapter of ACL’89*. 