Delayed interpretation, shallow processing and constructions: the basis of the "interpret whenever possible" principle

Philippe Blache

Delayed interpretation, shallow processing and constructions: the basis of the "*interpret whenever possible*" principle

Philippe Blache
CNRS & Aix-Marseille Université
Laboratoire Parole et Langage
blache@blri.fr

Abstract

We propose in this paper to investigate the "*interpret whenever possible*" principle which consists in delaying the processing mechanisms until enough information becomes available. This principle relies on the identification of elementary units called chunks, that are identified by means a basic features. These chunks are segments of the input to be processed. In some cases, depending on the accessibility of the information they bear, chunk can be linguistically structured elements. In other cases, they are simple segments. Chunks are stored in a buffer of the working memory and progressively grouped (on the basis of a cohesion measure) when possible, identifying progressively the different constructions of the input. The global interpretation of a linguistic input is then not based anymore on a word-by-word mechanism, but on the grouping of these constructions that are constitute the backbone of the "*interpret whenever possible*" principle.

1 Introduction

From different perspectives, natural language processing, linguistics and psycholinguistics shed light on the way humans process language. However, this knowledge remains scattered: classical studies usually focus on language processing subtasks (e.g. lexical access) or modules (e.g. morphology, syntax), without being aggregated into a unified framework. It remains then very difficult to find a general model unifying the different sources of information into a unique architecture.

One of the problems lies in the fact that we still know only little about how the different dimensions of language (prosody, syntax, pragmatics, semantics, etc.) interact. Some linguistic theories exist, in particular within the context of *Construction Grammars* [Fillmore, 1988, Goldberg, 2003, Blache, 2016], that propose approaches making it possible to gather these dimensions and implement their relations. These frameworks rely on the notion of *construction*, which is a set of words linked by specific properties at any level (lexical, syntactic, prosodic, etc.) and to which a specific meaning, which is often non transparent or accessible compositionnaly (e.g. idioms or multi-word expressions), can be associated. Interestingly, these theories also provide a framework for integrating multimodal information (verbal and non verbal). Interpreting a construction (i.e. accessing to its associated meaning) results from the interaction of all the different dimensions. In this organization, processing a linguistic production is not a linear process, but uses mechanisms for a global recognition of the constructions. Contrarily to incremental architectures (see for example

[Ferreira and Swets, 2002, Rayner and Clifton, 2009]), the syntactic, semantic, and pragmatic processing is not done word-by-word, but more globally, on the basis of such constructions.

This conception of language processing requires a *synchronization* procedure for the alignment of all the different sources of information in order to identify a construction and access to its meaning. In natural situations (e.g. conversations), the different input flows can be verbal (prosody, syntactic, pragmatics, etc.) and non verbal (gestures, attitudes, emotions, context, etc.), they are not temporally strictly synchronized. It is then necessary to explain how information can be temporarily stored and its evaluation delayed until enough information becomes available. In this perspective, the input linguistic flow (being it read or heard) is segmented into elements that can be of any form, partially or entirely recognized: segments of the audio flow, set of characters, but also when possible higher level segments made of words or even clusters of words. We address in this chapter these problems through several questions:

1. What is the nature of the delaying mechanism?
2. What is the nature of the basic units and how can they be identified?
3. How is the delaying mechanism implemented?

2 Delayed processing

Different types of delaying effects can occur during language processing. For example, at the brain level, it has been shown that language processing may be impacted by the presentation rate of the input. This phenomena has been investigated in [Vagharchakian et al., 2012] claiming that when the presentation rate increases and becomes faster than the processing speed, intelligibility can collapse. This is due to the fact that language network seems to work in a constant of time: cortical processing speed is shown by the authors to be tightly constrained and cannot be easily accelerated. As a result, when the presentation rate increases, the processing speed remaining constant, a blocking situation can suddenly occur. Concretely, this means that when the presentation rate is accelerated, and because the processing speed remains constant, a part of the input stream has to be buffered. Experiments show that the rate can be accelerated of 40% before reaching a collapse of intelligibility. This situation occurs when the buffer becomes saturated and is revealed at the cortical level by the fact that the activation of the higher-order language areas (that are said to reflect intelligibility [Friederici et al., 2010]) drops suddenly, showing that the input signal becomes unintelligible.

This models suggests that words can be processed immediately when presented at a slow rate, in which case the processing speed is that of the sensory system. However, when the rate increases and words are presented more rapidly, the processing speed limit is reached and words cannot be processed in real time anymore. In such a situation, words have to be stored in a buffer, from which they are retrieved in a *first-in-first-out* manner, when cognitive resources become available again. When the presentation rate is higher than the processing speed, the number of words to be stored increases. A lock occurs when the maximal capacity of the buffer is reached, entailing a collapse of intelligibility.

Besides this buffering mechanism, other cues indicate that the input is probably not processed linearly, word-by-word, but rather only from time to time. This conception means that even in normal cases (i.e. without any intelligibility issue), the interpretation is only done periodically, the basic units being stored before being processed. Several studies have

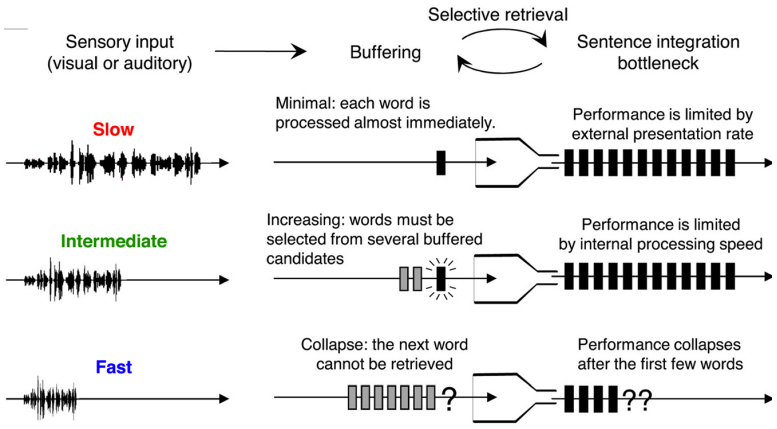


Figure 1: Illustration of the bottleneck situation, when presentation rate exceeds processing speed (reproduced from [Vagharchakian et al., 2012])

investigated such phenomenon. At the cortical level, the analysis of stimulus intensity fluctuation reveals the presence of specific activity (spectral peaks) after phrases and sentences [Ding et al., 2016]. The same type of effect can also be found in eye-movement during reading: longer fixations are observed when reading words that end a phrase or a sentence. This *wrap-up effect* [Warren et al., 2009], as well as the presence of different timescales at the cortical level described above, constitute cues in favor of a delaying mechanism in which basic elements are stored temporarily, and an integration operation is triggered when enough material becomes available for the interpretation.

At the semantic level, other evidences also show that language processing, or at least language interpretation, is not strictly incremental. Interesting experiences have been done revealing that language comprehension can stay very superficial: [Rommers et al., 2013] has shown that in an idiomatic context, the access to the meaning of words can be completely switched off, replaced by a global access at the level of idiom. This effect has been shown at the cortical level: when introducing a semantic violation within an idiom, there is no difference between hard and soft semantic violations (which is not the case in a comparable non idiomatic context): in some cases, processing a word does not mean integrating it into a structure. On the contrary, there is in this situation a simple shallow process scanning the word, without doing any interpretation. The same type of observation has been done in reading studies: depending on the task (for example when very simple comprehension questions are expected), the reader may apply a superficial treatment [Swets et al., 2008]. This effect is revealed by the fact that ambiguous sentences are read faster, meaning that no resolution is done and the semantic representation remains underspecified. Such variation in the level of processing depends then on the context: when the pragmatic and semantic context carries enough information, it renders the complete processing mechanism useless, the interpretation being predictable. At the attentional level, this observation is confirmed in [Astheimer and Sanders, 2009], showing that the allocation of attentional resources to certain time windows depends on its predictability: minimal attention is allocated when information is predictable or, on the contrary, maximal attention is involved in case of mismatch with expectations. The same type of variation is observed when listener adapts its perceptual strategy to the speakers, applying *perceptual accommodation* [Magnuson and Nusbaum, 2007].

These observations are in line with the *Good-enough Theory* [Ferreira and Patson, 2007] for which the interpretation of complex material is considered to be often shallow and incomplete. This model suggests that interpretation is only done from time to time, on the basis of a small number of adjacent words, and delaying the global interpretation until enough material becomes available. This framework and the evidences on which it relies also reinforce the idea that language processing is generally not linear and word-by-word. On the opposite, it can be only very shallow and when necessary delayed.

3 Working Memory

The delaying mechanism relies implicitly on a storage device which is implemented in the *short-term memory*, which is the basis of the cognitive system organization, by making it possible to store temporarily pieces of information of any nature. In general, it is considered that this memory is mainly devoted to storage. However, a specific short-term memory, called *working memory*, also allows for the manipulation of the information and a certain level of processing. It works as a buffer in which subpart of the information, that can be partially structured, is stored. Some models [Baddeley, 1986, Baddeley, 2000] proposes an architecture in which the working memory plays the role of a supervisor, on top of different sensory-motor loops as well as an episodic buffer.

One important feature of the working memory (and short-term memory in general) is its limited capacity. In a famous paper, [Miller, 1956] evaluated this limit to a "magic" number of seven units. However, it has been observed that units to be stored in this memory are not necessarily atomic, they can also constitute groups that are considered then as a single units. For example, stored elements can be numbers, letters, words, or even sequences, showing that groups can be encoded as a single units. In this case, the working memory stores not directly the set of elements, but more probably the set of pointers towards the location of the elements in another (lower) part of the short-term memory. These types of higher-level elements are called *chunks* which basically consist, in the case of language, as set of words.

Working memory occupies a central position in cognitive architectures such as ACT-R (*Adaptive Character of Thought-Rational*, see [Anderson et al., 2004]). In this model, short-term information (chunks) is stored into a set of buffers. The architecture, in the manner of that proposed by [Baddeley, 1986], is organized around a set of modules (manual control, visual perception, problem state, control state and declarative memory) coordinated by a supervising system (the production system). Each module is associated to a buffer which contains one chunk, defined as a unit containing a small amount of information. Moreover, in this organization, each buffer can contain only one unit of knowledge.

ACT-R has been applied to language processing, in which short-term buffers play the role of an interface between procedural and declarative memories (the different types of linguistic knowledge) [Lewis and Vasishth, 2005, Reitter et al., 2011]. Buffers contain chunks (information units) that are represented as lists of attribute-value pairs. Chunks are stored in the memory, they form a unit and they can be directly accessible, as a whole. Their accessibility depends on a level of *activation*, making it possible to control their retrieval in the declarative memory. Chunk's activation consists of several parameters: latency since its last retrieval, weights of the elements in relation with the chunk as well as the strength of these relations. It can be integrated into the following formula, quantifying the activation A of a chunk i:

$$A_i = B_i + \sum_j W_j S_{ji} \quad (1)$$

In this formula, B represents the basic activation of the chunk (its frequency and the recency of its retrieval), W indicates the weights of the terms in relation with i and S the strength of the relations linked other terms to the chunks. It is then possible to associate to a chunk its level of activation. The interesting point is that chunk activation is partially dependent from the context: the strength of the relations with other elements has a consequence on the level of activation, controlling its probability as well as the speed of its retrieval.

This architecture implicitly contains the idea of delayed evaluation: the basic units are first identified and stored into different buffers, containing pieces of information that can be atomic or structured. Moreover, this proposal also gives indications on the type of the retrieval. The different buffers in which chunks are stored is not implemented as a stack, following a *first-in-first-out* retrieval mechanism. On the contrary, chunks can be retrieved in any order, with a preference given first to that with the higher activation value.

The ACT-R model and the activation notion give a more precise account of comprehension difficulties. We have seen in the previous section that they can be the consequence of a buffer saturation (in computational terms, a *stack overflow*). Such difficulties are controlled thanks to the *decay of accessibility* of stored information [Lewis and Vasishth, 2005]. This explanation is complementary with observations presented in the previous section: the activation level has a correlation with the processing speed. Chunks with a high activation will be retrieved rapidly, decreasing the number of buffered elements. When many chunks have a low activation, the processing speed decreases, resulting in a congestion of the buffers.

One important question in this architecture is role of working memory in procedural operations, and more precisely the construction of the different elements to be stored. In some approaches, working memory plays a decisive role in terms of integration: basic elements (lexical units) are assembled into structured ones, in function of their activation. In this organization, working memory becomes the site where linguistic analysis is done. This is what has been proposed for example in the *"Capacity theory of comprehension"* [Just and Carpenter, 1992] for which working memory plays a double role of storage and processing. In this theory, elements of any level can be stored and accessed: words, phrases, thematic structures, pragmatic information, etc. It is however difficult to explain how such model can implement at the same time a delaying aspect (called "*wait-and-see*" by the authors) and an incremental comprehension system interpreting step-by-step. In their study on memory capacity, [Vagharchakian et al., 2012] propose a simpler view with a unique input buffer whose role is limited to storing words. In our approach, we adopt an intermediate position in which the buffer is limited to storage, but elements of different types can be stored, including partially structured ones such as chunks.

4 How to recognize chunks: the segmentation operations

The hypothesis of a delayed evaluation in language processing not only relies on a specific organization of the memory, but also requires a mechanism for the identification of the elements to be stored in the buffer. Two important questions are to be answered here: what is the nature of these elements, and how can they be identified. Our hypothesis relies on the idea that no deep and precise linguistic analysis is done at a first stage. If so, the question

is to explain and describe the mechanisms, necessarily at a low level, for the identification of the stored elements.

These questions are more generally related to the general problem of segmentation. Given an input flow (for example connected speech), what types of element can be isolated and how? Some mechanisms, specific to the audio signal, are at work in speech segmentation. Many works addressing this question [Mattys et al., 2005], [Goyet et al., 2010], [Newman et al., 2011], [Endress and Hauser, 2010] exhibit different cues, at different levels, that are used in particular (but not only) for word segmentation tasks, among which:

- *Prosodic level*: stress, duration, pitch information can be associated in some languages to specific positions in the word (for example initial or final), helping in detecting the word boundaries
- *Alophonic level*: phonemes are variable and their realization can depend on their position within words
- *Phonotactic level*: constraints on the ordering of the phonemes, which gives information about the likelihood that a given phoneme is adjacent to another one within and between words
- *Statistical/distributional properties*: transitional probabilities between consecutive syllables

Word segmentation results from the satisfaction of multiple constraints encoding different types of information such as phonetic, phonological, lexical, prosodic, syntactic, semantic, etc. (see [McQueen, 2010]). However, most of these segmentation cues are at a low level and do not involve an actual lexical access. In this perspective, what is interesting is that some segmentation mechanisms are not dependent from the notion of word and then can be also used in other tasks than word segmentation. This is very important because the notion of word is not always relevant (because involving rather high-level features, including semantic ones). In many cases, other types of segmentations are used, without involving the notion of words, but staying at the identification of larger segments (for example prosodic units), without entering into a deep linguistic analysis.

At a higher level, [Dehaene et al., 2015] has proposed to isolate five mechanisms making it possible to identify sequence knowledge:

- *Transition and timing knowledge*: when presenting a sequence of items (of any nature), at a certain pace, the transition between two items is anticipated thanks to the approximate timing of the next item.
- *Chunking*: contiguous items can be grouped into a same unit, thanks to the identification of certain regularities. A chunk is simply defined here in terms of a set of contiguous items that frequently co-occur and then can be encoded as a single unit.
- *Ordinal knowledge*: a recurrent linear order, independently from any timing, constitutes an information for the identification of an element and its position.
- *Algebraic patterns*: when several items have an internal regular pattern, their identification can be done thanks to this information.
- *Nested tree structures generated by symbolic rules*: identification of a complex structure, gathering several items into a unique element (typically a phrase)

What is important in these sequence identification systems (at least the first four of them) is the fact that they apply to any type of information and rely on low-level mechanisms, based on the detection of regularities and when possible their frequency. When applied to language, these systems explain how syllables, patterns or groups can be identified directly. For example algebraic patterns are specific to certain construction such as in the following example, taken from a spoken language corpus "*Monday, washing, Tuesday, ironing, Wednesday, rest*". In this case, without any syntactic or high-level processing, and thanks to the regularity of the pattern /*date - action*/, it is possible to segment the three subsequences and group them into a unique general one. In this case, a very basic mechanism, *pattern identification*, offers the possibility to identify a construction (and access directly to its meaning).

When putting together the different mechanisms described in this section, we obtain a strong set of parameters that offer the possibility to segment the input into units. In some cases, when cues are converging enough, the segments can be words. In other cases, they are larger units. For example, long breaks (higher than 200ms) are a universal segmentation constraint in prosody: two such breaks identifies the boundaries of segment (that can correspond to a prosodic unit).

As a result, we can conclude that several basic mechanisms, that do not involve deep analysis makes it possible to segment the linguistic input, being it read or heard. Our hypothesis is that these segments are the basic units stored initially in the buffers. When possible, the stored units are words, but not necessarily. In the general case, they are sequences of characters or phonemes that can be retrieved later. This is what occurs when hearing a speaker without understanding: the audio segment is stored and access later when other sources of information (for example the context) become available and make it possible to refine the segmentation into words.

5 The delaying architecture

Following the different elements presented so far, we propose to integrate the notion of delayed evaluation and chunking into the language processing organization. This architecture relies on the idea that the interpretation of a sentence (leading to its comprehension) is only done *whenever possible*, instead of word-by-word. The mechanism consists in accumulating enough information before any in-depth processing. Doing this means first the capacity to identify atomic units without making use of any deep parsing and second to store these elements and retrieve them when necessary.

We do not address here the question of building an interpretation, but focus only on this preliminary phase of accumulating pieces of information. This organization relies on a two-stage process distinguishing between a first level of packaging and a second corresponding to a deeper analysis. Such a level distinction can recall the well-known "*Sausage Machine*" [Frazier and Fodor, 1978] that distinghes a first phase called the *Preliminary Phrase Packager (PPP)*, consisting in identifying the possible groups (or chunks) in a limited window made of 6 or 7 words. In this proposal, the groups correspond to phrases, that can be incomplete. The second level is called the *Sentence Structure Supervisor (SSS)* and groups the units produced in the *PPP* into larger structures. In this classical architecture, each level involves a certain type of parsing, relying on grammatical knowledge. Moreover, the interpretation is supposed to be done starting from the identification of the syntactic structure, in a classical compositional perspective.

Our proposal also relies on a two-stage organization:

1. Segmenting and storing
2. Aggregating complex chunks

However, this model does not have any a priori on the type of units to be built: they are not necessarily phrases, they can be simply made of unstructured segments of the input. Moreover, the second stage is not obligatory: the recognition of a construction, and the interpretation of the corresponding subpart of the input, can be done at the first level.

We detail in the following these two stages, at the basis of the more general "*interpretation whenever possible*" organization.

5.1 Segment-and-store

The first stage when processing a linguistic input (text or speech) is the segmentation into atomic chunks. Atomic means here that no structure is built, chunks being only segments of the input, identified thanks to low-level parameters. In other words, no precise analysis of the input is done, the mechanism consisting in gathering all possible information available immediately. As a result, because the level of precision of the information can be very different, chunks can be of many different types and levels. Some of the segmentation mechanisms are indeed very general or even universal. For example, the definition of "*inter-pausal units*" relies on the identification of long breaks in the audio signal. The resulting chunk is a long sequence of phonemes without internal organization or sub-segmentation. In some (rare) cases, no other features than long breaks are available and the chunk remains large and stored as such. However, in most of the situations, more information is available, making it possible to identify finer chunks, and when possible words. Several such segmenting features exist, in particular:

- *Prosodic contours, stress*: pitch, breaks, duration, stress may indicate word boundaries.
- *Phonotactic constraints*: language-dependent constraint on the sequence of phonemes. The violation of such constraints may indicate boundaries.
- *Lexical frequency units*: in some cases, an entire unit can be highly predictable (typically very frequent words, named entities, etc.), making it possible to directly segment the input.

These features are subject to high variation and do not lead in all cases to a segmentation. When ambiguity is high, no finer segmentation is done at this stage. On the opposite, these low-level features can often lead to the possibility of segmenting into words. What is important is that these features correspond to information that can be directly assessed, independently from any other property or knowledge.

At this first stage, atomic chunks are stored into the buffers. We present in the following section the next step of this pre-processing phase, consisting in aggregating chunks.

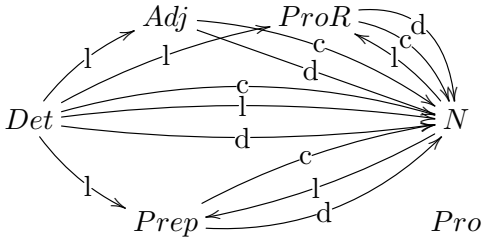
5.2 Aggregating by Cohesion

Constructions can be described as a set of interacting properties. This definition offers the possibility to conceive a measure based on the number of these properties and their weights,

as proposed in [Blache, 2016]. At the syntactic level, the set of properties describing a construction corresponds to a graph in which nodes are words and edges represent the relations. The graph density constitutes then a first type of measure: a high density of the graph corresponds to a high number of properties, representing then a certain type of cohesion between the words. Moreover, the quality of these relations can also be evaluated, some properties being more important than others (which is represented by their weighting). A high density of hard properties (i.e. with heavy weights) constitute then a second type of information. Finally, some sentences can be non-canonical, bearing certain properties that are violated (for example in case of agreement or linear precedence violation). Taking into consideration the number of violated properties in comparison with the satisfied ones is the last type of indication we propose to use in the evaluation of the cohesion.

Our hypothesis is that a correlation exists between the cohesion measure, defined on the basis of these three types of information, and the identification of a construction. In other words, a construction correspond to a set of words linked with a high number of properties, of heavy weights, with no or few violations.

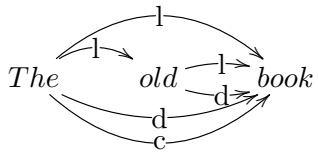
The first parameter of the cohesion measure relies on the number of properties that are assessed for a given construction, in comparison with the possible properties in the grammar. The following graph illustrates the set of properties *in the grammar* describing the nominal construction[1]:



(2)

The number of possible relations in which a category is involved can be estimated by the number of incident relations of the corresponding vertex in the graph (called in graph theory the *vertex* degree). We propose then to define the degree of a category by this measure. In the previous graph, we have the following degrees: $deg_{[gram]}(N) = 9; deg_{[gram]}(ProR) = 2; deg_{[gram]}(Adj) = 1$.

During a parse (i.e. knowing the list of categories), the same type of evaluation can be applied to the constraint graph describing a construction, as in the following example:



Each word is involved into a set of relations. The degree of a word is, similarly to the grammar, the set of incident edges of a word. In this example, we have: $deg_{[sent]}(N) = 5; deg_{[sent]}(Adj) = 1; deg_{[sent]}(Det) = 0$.

[1]The letters d, l, c stand respectively for *dependency, linearity* and *cooccurrence* properties.

The first parameter of our estimation of the cohesion relies on a comparison of these two values: for a given word, we know from the grammar the number of properties in which it could theoretically be involved. We also know from the parsing of a given sentence how many of these properties are effectively assessed. We can define then a value, the *completeness ratio*, indicating the density of the category: the higher the number of relations in the grammar is verified, the higher the completeness value:

$$Comp(cat) = \frac{deg_{[sent]}(cat)}{deg_{[gram]}(cat)}$$

Besides this completeness ratio, it is also interesting to examine the density of the constraint graph itself. In graph theory, this value is calculated as a ratio between the number of edges and the number of vertices. It is more precisely defined as follows (S is the constraint graph of a sentence, E the set of edges, V the set of vertices):

$$Dens(S) = \frac{|E|}{5 * |V|(|V| - 1)}$$

In this formula, the numerator is the number of existing edges, the denominator is the total number of possible edges (each edge connecting two different vertices, multiplied by 5, the number of different types of properties). This value makes it possible to distinguish between *dense* vs. *sparse* graphs. In our hypothesis, a dense graph is correlated with a construction.

The last parameter taken into account is more qualitative and takes into account the weights of the properties. More precisely, we have seen that all properties can be either satisfied or violated. We define then a normalized satisfaction ratio as follows (where W^+ is the sum of the weights of the satisfied properties and W^- that of the violated ones):

$$Sat(S) = \frac{W^+ - W^-}{W^+ + W^-}$$

Finally, the cohesion value can be calculated as a function of the three previous parameters as follows (C being a construction, G_C its corresponding constraint graph):

$$Cohesion(C) = \sum_{i=1}^{|S|} Comp(w_i) * Dens(G_C) * Sat(G_C)$$

Note that the *density* and *satisfaction* parameters can be evaluated directly, without depending on the context and without needing to know the type of the construction. On the contrary, evaluating the *completeness* parameter requires to know the construction in order to extract from the grammar all the possible properties that describe it. In a certain sense, the two first parameters are *basic*, in the same sense as described for properties, and can be assessed automatically.

The *cohesion* measure offers a new estimation of the notion of *activation*. Moreover, it also provides a way to directly identify constructions on the basis of simple properties. Finally, it constitutes an explicit basis for the implementation of the general parsing principle stipulating that constructions or chunks are set of words with a high density of relations of heavy weights. This definition corresponds to the *Maximize on-line Processing* principle [Hawkins, 2003]

which stipulates that *"the human parser prefers to maximize the set of properties that are assignable to each item X as X is parsed. [...] The maximization difference between competing orders and structures will be a function of the number of properties that are misassigned or unassigned to X in a structure S, compared with the number in an alternative."*

This principle offers a general background of our conception of language processing. Instead of building a syntactic structure serving as support of the comprehension of a sentence, the mechanisms consists in a succession of chunks, maximizing the cohesion function estimated starting from the available information. When the density of information (or the cohesion) reaches a certain threshold, the elements can be grouped into a unique chunk, stored in the working memory. When the threshold is not reached, the state of the buffer is not modified and a new element of the input stream is scanned. This general parsing mechanism offers the possibility to integrate different sources of information when they become available by delaying the evaluation, waiting until a certain threshold of cohesion can be identified. This constitutes a framework for implementing the basic processing of the good-enough theory: *interpret whenever possible.*

6 Conclusion

Understanding language is theoretically a very complex process, involving many different sources of information. Moreover, it has to be done in real-time. Fortunately, in many cases, the understanding process can be facilitated thanks to different parameters: predictability of course, but also the fact that entire segments of the input can be processed directly. This is the case of most of the *constructions*, in which the meaning can be accessed directly, the construction being processed as a whole. At a lower level, it is also possible to identify subparts of the input (for example patterns, prosodic units, etc.) from which global information can be retrieved directly. Different observations show that low-level features usually makes it possible to identify such global segments. The language processing architecture we propose in this paper relies on this: instead of recognizing words and then trying to integrate them step-by-step into a syntactic structure to be interpreted, segments are first identified. These segments can be of any type: sequences of phonemes, words, group of words, etc. Their common feature is that they do not need any deep level information or process to be recognized.

Once the segments (called *chunks*) identified, they are stored into a buffer, without any specific interpretation. In other words, the interpretation mechanism is *delayed* until enough information becomes available. When a new chunk is buffered, an evaluation of its *cohesion* with the existing ones in the buffer is done. When the cohesion between different chunks (that corresponds to the notion of activation in cognitive architectures) reaches a certain threshold, they are merged into a unique one, replacing them in the buffer, as a single unit. This mechanism makes it possible to progressively recognize constructions and access directly to their meaning.

This organization, instead of a word-by-word incremental mechanism, implements the *"interpret whenever possible"* principle. It constitutes a framework for explaining all the different delaying and shallow processing mechanisms that has been observed.

References

[Anderson et al., 2004] Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., and Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, 111(4):1036–1060.

[Astheimer and Sanders, 2009] Astheimer, L. B. and Sanders, L. D. (2009). Listeners modulate temporally selective attention during natural speech processing. *Biological Psychology*, 80(1):23–34.

[Baddeley, 1986] Baddeley, A. (1986). *Working Memory*. Oxford: Clarendon Press.

[Baddeley, 2000] Baddeley, A. D. (2000). The episodic buffer: a new component of working memory? *Trends in Cognitive Sciences*, 4(11):417–423.

[Blache, 2016] Blache, P. (2016). Representing syntax by means of properties: a formal framework for descriptive approaches. *Journal of Language Modelling*.

[Dehaene et al., 2015] Dehaene, S., Meyniel, F., Wacongne, C., Wang, L., and Pallier, C. (2015). The neural representation of sequences: From transition probabilities to algebraic patterns and linguistic trees. *Neuron*, 88(1).

[Ding et al., 2016] Ding, N., Melloni, L., Zhang, H., Tian, X., and Poeppel, D. (2016). Cortical tracking of hierarchical linguistic structures in connected speech. *Nature Neuroscience*, 19(1):158–164.

[Endress and Hauser, 2010] Endress, A. D. and Hauser, M. D. (2010). Word segmentation with universal prosodic cues. *Cognitive Psychology*, 61(2).

[Ferreira and Patson, 2007] Ferreira, F. and Patson, N. D. (2007). The 'good enough' approach to language comprehension. *Language and Linguistics Compass*, 1(1).

[Ferreira and Swets, 2002] Ferreira, F. and Swets, B. (2002). How incremental is language production? evidence from the production of utterances requiring the computation of arithmetic sums. *Journal of Memory and Language*, 46.

[Fillmore, 1988] Fillmore, C. J. (1988). The mechanisms of "construction grammar". In *Proceedings of the Fourteenth Annual Meeting of the Berkeley Linguistics Society*, pages 35–55.

[Frazier and Fodor, 1978] Frazier, L. and Fodor, J. D. (1978). The sausage machine: A new two-stage parsing model. *Cognition*, 6(4):291–325.

[Friederici et al., 2010] Friederici, A., Kotz, S., Scott, S., and Obleser, J. (2010). Disentangling syntax and intelligibility in auditory language comprehension. *Human Brain Mapping*, 31(448).

[Goldberg, 2003] Goldberg, A. E. (2003). Constructions: a new theoretical approach to language. *Trends in Cognitive Sciences*, 7(5):219–224.

[Goyet et al., 2010] Goyet, L., de Schonen, S., and Nazzi, T. (2010). Words and syllables in fluent speech segmentation by french-learning infants: An erp study. *Brain Research*, 1332(C).

[Hawkins, 2003] Hawkins, J. (2003). Efficiency and complexity in grammars: Three general principles. In Moore, J. and Polinsky, M., editors, *The Nature of Explanation in Linguistic Theory*, pages 95–126. CSLI Publications.

[Just and Carpenter, 1992] Just, M. A. and Carpenter, P. A. (1992). A capacity theory of comprehension: Individual differences in working memory. *Psychological Review*, 99(1):122–149.

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

[Magnuson and Nusbaum, 2007] Magnuson, J. and Nusbaum, H. (2007). Acoustic differences, listener expectations, and the perceptual accommodation of talker variability. *Journal of Experimental Psychology: Human Perception and Performance*, 33(2):391–409.

[Mattys et al., 2005] Mattys, S. L., White, L., and Melhorn, J. F. (2005). Integration of multiple speech segmentation cues: A hierarchical framework. *Journal of Experimental Psychology*, 134(4).

[McQueen, 2010] McQueen, J. M. (2010). Speech perception. In Lamberts, K. and Goldstone, R., editors, *The handbook of cognition*. London: Sage.

[Miller, 1956] Miller, G. (1956). The magical number seven, plus or minus two: some limits on our capacity for processing information. *Psychological Review*, 63(2):81–97.

[Newman et al., 2011] Newman, R. S., Sawusch, J. R., and Wunnenberg, T. (2011). Cues and cue interactions in segmenting words in fluent speech. *Journal of Memory and Language*, 64(4).

[Rayner and Clifton, 2009] Rayner, K. and Clifton, C. (2009). Language processing in reading and speech perception is fast and incremental: Implications for event related potential research. *Biological Psychology*, 80(1).

[Reitter et al., 2011] Reitter, D., Keller, F., and Moore, J. D. (2011). A computational cognitive model of syntactic priming. *Cognitive Science*, 35(4):587–637.

[Rommers et al., 2013] Rommers, J., Dijkstra, T., and Bastiaansen, M. (2013). Context-dependent Semantic Processing in the Human Brain: Evidence from Idiom Comprehension. *Journal of Cognitive Neuroscience*, 25(5):762–776.

[Swets et al., 2008] Swets, B., Desmet, T., Clifton, C., and Ferreira, F. (2008). Underspecification of syntactic ambiguities: Evidence from self-paced reading. *Memory and Cognition*, 36(1):201–216.

[Vagharchakian et al., 2012] Vagharchakian, L., G., D.-L., Pallier, C., and Dehaene, S. (2012). A temporal bottleneck in the language comprehension network. *Journal of Neuroscience*, 32(26):9089–9102.

[Warren et al., 2009] Warren, T., White, S. J., and Reichle, E. D. (2009). Investigating the causes of wrap-up effects: Evidence from eye movements and e–z reader. *Cognition*, 111(1):132–137.