Linguistic Units from a Left-Corner Episodic Grammar

Michael Anthony Cabot
6047262

Bachelor thesis
Credits: 6 EC

Bachelor Opleiding Kunstmatige Intelligentie

University of Amsterdam
Faculty of Science
Science Park 904
1098 XH Amsterdam

Supervisor
dr. G. Borensztajn

Institute for Language, Logic and Computation (ILLC)
Faculty of Science
University of Amsterdam
Science Park 904
1098 XH Amsterdam

June 26, 2012

1 Introduction

This paper is driven by my curiosity in understanding how children learn their first natural language. A child acquiring her first language is faced with the challenge of representing and producing utterances from an unknown grammar. The internal representation of the grammar she learns must reflect one of language's highly characteristic properties, its hierarchical nature [8]. Another characteristic feature of language is its productivity, i.e. a child must be able to produce an unbound number of novel phrases from a limited amount of words and rules.

In order to understand *how* a child is able to learn such a grammar it is first important to understand *what* a child is learning. My search for understanding how children learn grammar has lead to the following two subquestions which will make up the focus of this paper:

- What are the linguistic units that children learn?
- How are these linguistic units encoded in the brain?

In section 2 I will discuss studies by Tomasello et al. that give empirical evidence of the types of phrases and constructions that children learn. Section 3 discusses the DOP model developed by Scha and Bod. This model gives an explanation of how these constructions can be represented using tree structures. Section 4 describes the episodic grammar: a cognitively plausible model that explains how tree structures can be encoded in the brain based on the relationship between the semantic and episodic memory system. Section 5 expands this model with the left-corner-parser: a cognitively plausible parsing strategy. Section 6 shows the difference between DOP and the episodic grammar. Section 7 shows a quantitative analysis of the three types of tree structures that occur in the Brown corpus, an annotated dataset provided by my supervisor, Gideon Borensztajn. The remainder of this paper discusses my findings, their significance and whether the left-corner episodic parser is suitable for modeling the way children parse language.

2 Usage-based approach

An important aspect of doing research on grammar acquisition is, or ought to be, understanding how operations on linguistic units result in utterances. The first step in doing so is to define the linguistic units; how they are represented and how they are learned. Usage-based models suggest that these units are acquired by inductive learning; A child learns a grammar by generalizing over utterances she has heard. Research by Tomasello [15] on usage-based models states that the utterance is the primary linguistic unit that children use to communicate. Once a child learns that the jiberish sounds her parents make are meant to manipulate her attention, she will attempt to reproduce these same sounds in order to shift her parents' attention to attend her own needs. A child will try to reproduce the whole utterance she has heard, but initially she will only succeed in producing one linguistic element out of the adult's whole utterance. For instance, if a child would like to ask '*Mother would you be so kind as to hand me the ball?*', you will only hear her say '*ball!*'. The case where one or a few

words function as a whole utterance is known as a *holophrase*. By the time a child produces multi-word phrases she tends to structure the utterance around one word or phrase that determines the function of the whole utterance, while the rest of the phrase (e.g. the subject or object) function as variable slots. For example, '*Where's the X?*', '*I wanna X*' and '*Throw X*'. Such a multi-word phrase is known as a *schema*: a phrase that contains one or more open slots which can be filled with a range of different lexical items.

Studies by Tomasello and Brooks [14, 16] show that these schemas tend to revolve around specific verbs and that initially a child does not generalize well over these different verb-schemas, thus forming so called verb-island constructions. This indicates that in early stages children mostly use language the way they have heard adults using it. Tomasello's research [15] shows that almost all novel utterances a child produces can be traced back to be combinations of phrases she has heard or said before. At some point children are able to create novel utterances. This can be done by filling in the open-slots in schemas. Over time these schemas grow more abstract, containing more open-slots and allowing more abstract categories to be places in these slots. This consequently allows a child to produce more creative novel utterances.

The following section will describe how the aforementioned schemas can be represented using a tree structure and how operations on these tree structures, similar to filling in a schema's open-slots, can produce novel phrases.

3 Data Oriented Parsing

In a Tree Substitution Grammar (TSG) the primary linguistic units are represented as parse tree fragments of arbitrary size and depth, where each non-terminal leaf node can be substituted by the root of another construction, eventually creating a complete derivation of a parse tree.

Data Oriented Parsing (DOP), a model developed by Bod and Scha [3, 11, 12], is a probabilistic TSG. A DOP tree fragment is valid when it has a single root and all of its nodes include all or none of their children. Two tree fragments can be combined by means of the substitution operator $\circ$ if the left-most non-terminal leaf node of the first fragment is identical to the root node of the second fragment. Figure 1 shows how three tree fragments are combined to create a derivation of the sentence *Sue takes her hat off*. A derivation of a sentence in DOP is a sequence of substitutions $t_1 \circ t_2 \circ ... \circ t_n$ where the final tree has as root the special syntactic category S and all the leaves are bound to the words in the sentence.

When calculating the probability of a derivation it is assumed that the probability of each substitution is context independent. The probability of a derivation is therefore the product of the used fragments conditioned on their roots:

$$P(t_1 \circ t_2 \circ ... \circ t_n) = \prod_n P(t_n | R(t_n)) \tag{1}$$

where $P(t_n|R(t_n))$ is the probability of subtree t_n, whose root $R(t_n)$ matches the left-most non-terminal leaf of the derivation so far. Because there can be different derivations that result in the same final tree (see figure 2), the probability of a parse is not equal to that of a derivation, but to the sum of all the

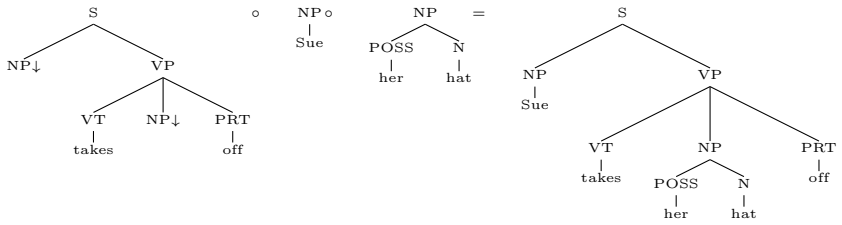


Figure 1: Derivation for *Sue takes her hat off* [5]. Non-terminal leaves are marked with ↓. Note that a free fragment always substitutes the left-most non-terminal leaf, thus creating the derivation *Sue takes her hat off* and not *Her hat takes Sue off*.

derivations that produce the same final parse tree:

$$P(T) = \sum_{d} \prod_{n} P(t_{nd}|R(t_{nd})) \tag{2}$$

where $P(t_{nd}|R(t_{nd}))$ is the probability of tree fragment n in derivation d.

The earliest estimator of calculating the probability of a tree fragment $P(t|R(t))$, known as DOP1 by Bod [1], uses the relative frequency condition on the root $R(t)$ of the tree:

$$P(t|R(t)) = \frac{|t|}{\sum_{t':R(t')=R(t)} |t'|} \tag{3}$$

where $|t|$ is the frequency of tree fragment t and $\sum_{t':R(t')=R(t)} |t'|$ is the frequency of all the tree fragments that have the same root as t. Since then other estimators have been developed.

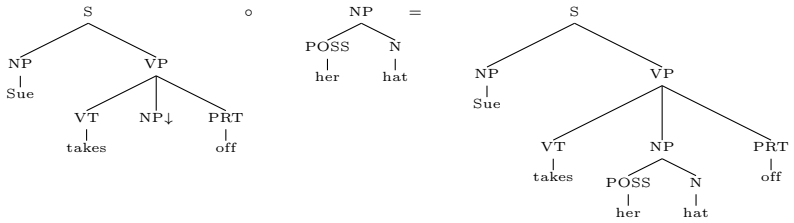


Figure 2: Alternative derivation for *Sue takes her hat off* compared to figure 1. Non-terminal leaves are marked with ↓.

Given a corpus of sentences annotated with syntactical parse trees, DOP will use these parse trees to create a fragment grammar: the set of valid tree fragments extracted from all the parse trees along with their occurring frequency. Figure 3 shows a selection of the valid tree fragments from an annotated sentence. The tree fragments in the fragment grammar are then combined to create derivations for a set of unannotated sentences. It is not necessary that all unannotated sentences occur in the annotated corpus, since tree fragments can be combined to create novel derivations.

When using a fragment grammar to parse a sentence, it will often be the case that a parse tree can be constructed in more ways than one. This could be because a sentence is ambiguous (see figure 4) or simply because the fragment

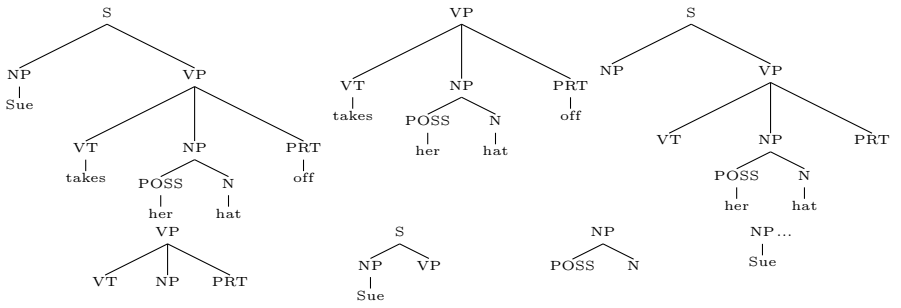


Figure 3: Some of the valid tree fragments taken from the parse tree of *Sue takes her hat off* (see figure 1).

grammar allows a sentence to be parsed with different tree structures (see figure 5). This means that a choice must be made which of the tree structures is best; the best one being the parse tree that is identical or most similar to the sentence's manually annotated parse tree.

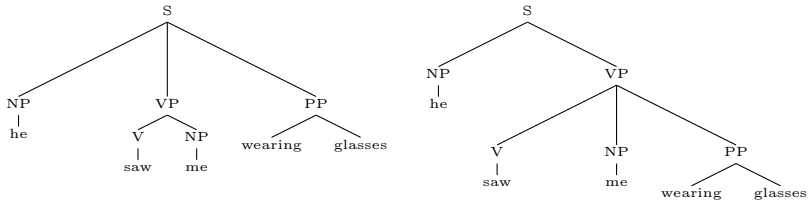


Figure 4: Two possible interpretations of the sentence *He saw me wearing glasses.*

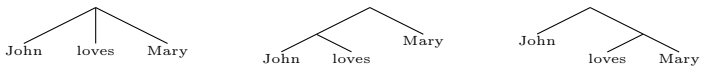


Figure 5: Three different tree structures that capture the sentence *John loves Mary.* The structure on the right is the sentence's true parse tree.

There are different approaches for choosing a derivation for a sentence. One approach is to choose the derivation that has the maximum probability according to equation (2). In this case the probability fully relies on the distribution of the probability mass over the tree fragments (e.g. according to equation (3)). Another approach is to choose the derivation with the least amount of tree fragments, i.e. the *shortest derivation* [2]. When choosing the shortest derivation there will be a bias towards using larger fragments. This approach is cognitively motivated by the principle of least effort: when encountering a novel sentence, the similarity with previously encountered sentences is attempted to be maximized. Note that choosing the derivation with the highest probability also induced a bias towards a derivation with few fragments. The more fragments that are used the more likely it is that the product of the probabilities of these fragments is small. The shortest derivation approach simply ensures that the least amount of fragments is always chosen.

When there are multiple derivations that have the shortest derivation, the

tie is broken according to their *rank* [13]. First the tree fragments in the fragment grammar are sorted by their frequency. The fragment with the highest frequency will have rank 0, the second highest rank 1, etc. The rank of each derivation is then calculated by summing the ranks of its tree fragments. The shortest derivation with the lowest rank is then chosen to be the derivation of a sentence. In the case that there are multiple shortest derivations that have an equal ranking score, one derivation is chosen at random.

DOP shares many similarities with Tomasello's usage-based model. DOP is usage-based, since it extracts its tree fragments from a corpus of language events, thus making it learn from experience; similar to Tomasello's schemas, DOP's tree fragments capture different utterance-lengths and different levels of abstraction; and the construction of novel phrases by filling in a schema's open-slots with other constructions is similar to DOP's substitution operator $\circ$.

The following section will describe a cognitive approach of building a usage-based model that is inspired by the shortest-derivation-DOP model.

4 Episodic Grammar

Our ability to acquire a language partially relies upon the declarative memory system, which is involved in representing conceptual knowledge and personal experiences. Declarative memory is, since Tulving [17], further divided into the following two components:

Semantic memory. A person's memory of general world knowledge in the form of concepts that are systematically related to one another.

Episodic memory. A person's memory of personally experienced events or *episodes*, that are embedded in a temporal, spatial and emotional context.

It is assumed that episodic and semantic memory are closely related; one common view is that an episodic memory is constructed as a set of pointers that bind together items stored in semantic memory. Borensztajn et al. [5] note that the division of declarative memory can also be found in the different approaches of grammar modeling. On one side the focus lies on finding empirical evidence for abstract rule-based grammars (e.g. context-free-grammar rules, such as S $\rightarrow$ NP VP), while on the other side the focus lies on the usage-based nature of grammar in which linguistic units are extracted from experience. Borensztajn et al. note that:

> [The] rule-based grammars are best thought of as instances of semantic memory, since they encode abstract, relational linguistic knowledge. The [usage-based] approach, on the other hand, suggests a role for episodic memory in sentence processing.

DOP is a model that attempts to bridge the gap between the rule-based and usage-based approach by using tree fragments as linguistic units that can range from context-free-rules to whole phrase structures. Borensztajn's *episodic grammar* [5] tries to bridge the gap between the two approaches even further by taking into account the following basic empirical facts about episodic memory:

Physical traces. All episodic memories that can be remembered must leave physical memory traces in the brain.

Sequentiality. Physical traces belonging to the same episode must bound together a sequence of semantic elements within a certain context.

Content addressability. Priming semantic memory can trigger an episodic memory with which it is associated by recalling the sequence of traces that belong to the same episode. To account for content addressability an episodic memory must support local access from semantic memory units to their associated episodes.

Separability and identifiability. It must be possible to disambiguate episodic traces that overlap with each other.

An example of an episodic memory, or *episode*, is my morning at the dentist. I lay in the chair, opened my mouth, got told a wisdom tooth was going to be removed and finally went home with a sedated jaw. Since I am able to recall this event it must mean that my mind contain traces that link *going to the dentist* with *having a sedated jaw*. The fact that I know that my jaw was sedated after I went to the dentist shows that there is a sequence that binds these traces. Furthermore, I am able to tell apart the time I went to the dentist and got my wisdom tooth removed from the other time I went to the dentist and everything was fine. This episode is accompanied by semantic memories of concepts such as the dentist, a wisdom tooth, sedation and all the things I associate with them. In the same manner this relation between semantic and episodic memory can be applied to language. The semantic memory would encode the linguistic structures that make up a phrase and the episodic memory binds these structures together.

Episodic grammar applies this idea of sequentially ordered unambiguous events to a parse tree. When parsing a sentence the path traversed through the tree is remembered by leaving traces in every node that is encountered. Each trace contains a *sentence-number*, i.e. the sentence being parsed when the trace is placed, and a *sequence-number*, i.e. the step of parsing in which the trace is placed. When encountering such a trace, the original parse tree can be reconstructed by following the path of traces that have the same sentence-number and have a connecting sequence-number. A node involved in multiple parses has multiple unique traces and can thus not cause an ambiguous episodic event. The sequentiality of the event is preserved by the sequence-numbers of the trace.

The episodic grammar represents a parse tree as a network of syntactic processing units, known as *treelets*. For simplicity, it may be assumed that a treelet consists of a context-free rule (e.g. S $\rightarrow$ NP VP) and a set of traces. Treelets can be combined to parse a sentence similarly to context-free rules, i.e. by means of substitution. Yet whereas a context-free grammar represents a parse with a tree construction (see figure 6), the episodic grammar represents a parse with a set of treelets that are connected by their traces. Figure 7 shows the top-down derivation of '*girl who dances likes tango*' (orange trances) and '*boy likes mango*' (blue traces). The treelets (the triangles and squares) represent the semantic knowledge of the model, while the traces (the colored ovals) connecting the treelets represent the episodic knowledge of the model. The figure shows how the parse tree of '*girl who dances likes tango*' can be reconstructed by following the arrows connecting the orange traces, starting at

the treelet with root S. Note that since the treelet NP → N is used twice in the derivation of this sentence, the treelet contains two orange traces.

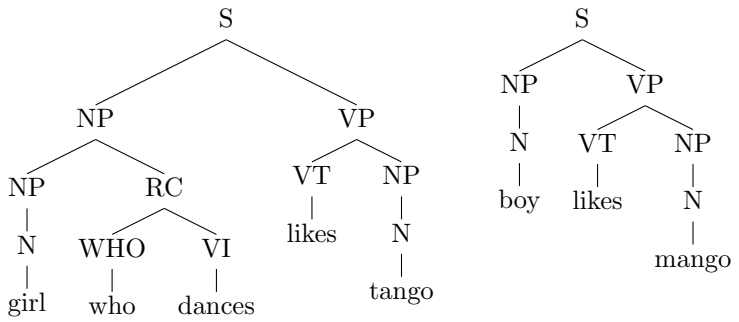


Figure 6: Parse tree of '*girl who dances likes tango*' and '*boy likes mango*'.

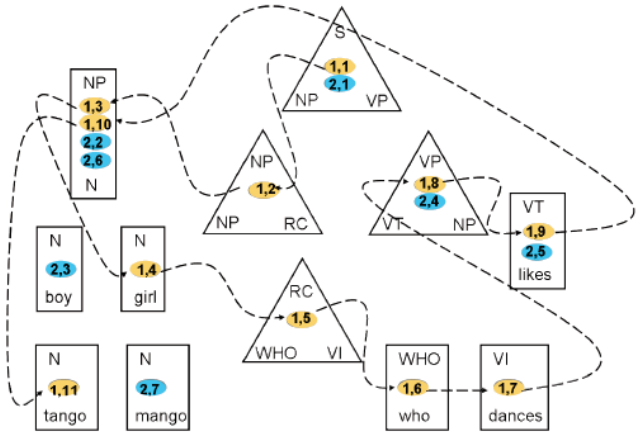
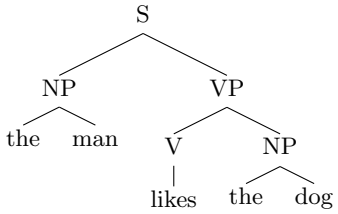


Figure 7: Treelets containing the traces of '*girl who dances likes tango*' (orange ovals) and '*boy likes mango*' (blue ovals). Each trace contains a sentence-number and a sequence-number. The arrows show the top-down derivation of the sentence '*girl who dances likes tango*'.

The episodic grammar can be trained using a corpus of sentences annotated with parse trees that contain traces. First, a treelet is created for every unique context-free rule that occurs in the corpus. Secondly, from each parse the derivation is extracted by turning a parse tree into a sequence of treelets ordered according to a chosen derivation strategy. Figure 8 shows how a parse tree is sequentialized with a top-down derivation strategy. Finally, these sequences are used to place traces in the treelets. Every step k in the derivation of sentence number s leaves a trace in the visited treelet, encoded as $\langle s, k \rangle$.

After training the episodic grammar the model can be used to parse new sentences. Just as with DOP there can be multiple derivations to parse a sentence and multiple techniques to choose one of these derivations. Just as DOP, the episodic grammar has a version that chooses the most probable derivation and



Derivation: ⟨S→NP VP, NP→the man, VP→V NP, V→likes, NP→the dog⟩

Figure 8: The derivation of the sentence '*the man likes the dog*' is extracted by sequentializing its parse tree using a top-down derivation strategy.

the shortest derivation. The shortest derivation approach follows the principle of least effort. This means the parser will follow the path it is most familiar with. The following describes how the episodic grammar chooses a derivation according to the shortest derivation length (SDL). This model is known as the *shortest derivation episodic grammar* (SD-EG). SD-EG will also be used throughout the rest of this paper.

Given an ongoing derivation d of a sentence that has arrived at a certain treelet t_q, one defines the cost of continuing the derivation to any other treelet $t_{q'}$ according to their traces that originate from derivations constructed during training. The shortest derivation length (SDL) defines the amount of different derivation fragments that are used when parsing a new sentence. This translates to incrementing the derivation length with 1 if $t_{q'}$ does not contain a trace that is connected with a trace in t_q. If $t_{q'}$ does contain a connecting trace, then the derivation length stays the same. Since the episodic grammar tries to offer an explanation of how tree structures can be represented in the brain, it is important that all computations are done locally. This is taken into account, since the current SDL is passed on to every treelet that is added to the derivation. The cost of adding a treelet to a derivation is thus calculated incrementally as follows:

$$\mathrm{SDL}(t_{q'}) = \begin{cases} \mathrm{SDL}(t_q) & \text{if } t_q \text{ contains trace } e \equiv \langle s, k-1 \rangle \\ & \text{and } t_{q'} \text{ contains trace } e' \equiv \langle s, k \rangle \\ \mathrm{SDL}+1 & \textit{else} \end{cases} \tag{4}$$

where $\mathrm{SDL}(t_q)$ is the current derivation length in treelet t_q and $\mathrm{SDL}(t_{q'})$ is the new derivation length in treelet $t_{q'}$.

5 Left-Corner-Parser

To further improve the episodic grammar of being a cognitively plausible model, a parsing strategy must be chosen that matches a human parser as best as possible. The human parser refers to the parsing strategy that the brain uses to process a sentence. A study by Crocker [9] shows that one of the most salient properties of the human parser is that is appears to operate incrementally. Both the top-down-parser (TDP) and bottom-up-parser (BUP) process a sentence incrementally from left-to-right. However, the TDP is not fully data-driven, since it first attempts to construct a large portion of a parse tree before it even

looks at the words in a sentence. Depending on the complexity of the sentence's grammar it may take a considerable time before a TDP chooses the correct upper-structure of the parse tree that will match with the words. BUP, on the other hand, is fully data-driven, but leaves adjacent constituents unresolved for an arbitrary period of time. For example, parsing the sentence '*boy likes mango*' (see figure 6) using a BUP will leave the category NP above subject *boy* unconnected to VP until the entire VP branch has been parsed. The standard assumption is that a parse cannot be interpreted until these constituents are connected. This means that BUP causes a delay in interpretation that is not likely to be present in the human parser.

The left-corner-parser (LCP) is a parsing strategy that combines the TDP and the BUP, thus creating a parser that is both incremental and data-driven. The *left corner* refers to the left-most symbol on the right hand side of a context-free-rule. For example, *NP* is the left-corner of the context-free-rule $S \rightarrow NP\ VP$. The LCP starts a parse at the leftmost word a sentence and uses the following operations to parse the rest:

Project. A treelet whose right-most child has been completed, i.e. all of its children have been parsed, can project upwards if it matches the left-corner of a context-free-rule.

Shift. If the right-most child of a treelet g has not been parsed, then the parser shifts downwards to the next word in the sentence. Since treelet g has not been completed, g becomes a goal treelet which the parse needs to return to at later stage of the parse. A stack is used to store these goal treelets since the last treelet added to the stack should be reached first in the parse [1].

Attach. When a shift is performed, the goal of the parse is to find a path back to this goal treelet by means of projections. An attach is a project-operation where the current node 'fits' with the goal node on top of the stack. This means that the left-hand side of the context-free-rule matches the left-most child of a goal treelet on top of the stack that has not yet been parsed. When an attach is successful the node is popped from the stack. If the current node is the right-most child of the goal node, then the parse can continue projecting upwards. If not, then another shift-operation is performed. An attach is always given priority over a project.

Figure 9 shows the path of a left-corner-parser through the parse tree of the sentence *girl who dances likes tango*. The parse starts with the left-most word *girl* and projects upwards until it reaches the treelet NP → NP RC. Since RC, the right-most child of this treelet, has not yet been parsed, the LCP shifts towards the next word in the sentence (i.e. *who*) and the treelet NP → NP RC is placed on top of the stack. Another shift occurs when the parse reaches the treelet RC → WHO VI, which is also placed on top of the stack. An attach occurs when the LCP notices that VI matches the left-most child of the treelet on top of the stack that has not yet been parsed. After attaching the treelet VI → dances, the treelet RC → WHO VI is completed and can continue projecting upwards. The parse is finished when the right-most child of the special node S has been attached.

[1] A stack follows the Last In First Out (LIFO) principle

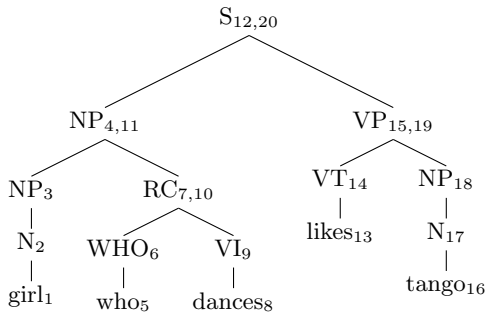


Figure 9: The numbers in subscript describe the path the LCP takes through the parse tree of '*girl who dances likes tango*'. These are equivalent to the sequence-numbers of a LCP derivation. Note that treelets with multiple children contain multiple sequence-numbers, since it is visited multiple times. First by means of projections and afterwards by means of attachments.

The episodic grammar can easily integrate any parsing strategy by changing the strategy of extracting a derivation from a parse tree. Integrating the LCP would thus mean that the traces of an episode follow the path of a left-corner parser. Figure 10 shows an episodic grammar using a left-corner derivation strategy. Note that the traces in the treelets correspond with the left-corner path shown in figure 9.

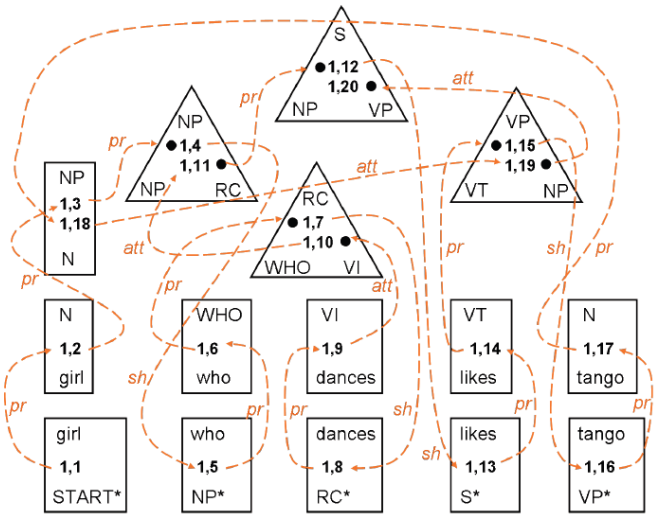


Figure 10: Treelets containing the traces of the sentence '*girl who dances likes tango*'. Each trace contains a sentence-number and a sequence-number. The arrows show the left-corner derivation of this sentence. The operations that bind the treelets are indicated with *pr* for project, *sh* for shift and *att* for attach.

The next section will give a qualitative analysis of the types of tree fragments that can be constructed by a left-corner episodic grammar.

6 Differences between DOP and the Episodic Grammar

This section will cover some of the most important differences between DOP and the episodic grammar (EG).

6.1 Derivation

Tree fragments in DOP exist explicitly as fixed tree structure. A derivation in DOP is defined as a sequence of such tree fragments, where the left-most non-terminal leaf is substituted by the root of the next tree fragment in the sequence, only if its root is equal to the left-most non-terminal leaf. In EG a derivation is defined as a set of treelets that are connected by their traces. A tree fragment in EG is thus not encoded as an explicit tree structure, but instead as an implicit one and can only be reconstructed by following the path of the traces that belong to the same derivation.

Since substitution in DOP only demands that the root of a tree fragments is equal with the left-most non-terminal leaf, the probability of substituting a tree fragment is conditioned on its root. This makes each substitution insensitive to the rest of the derivation. EG, on the other hand, incrementally expands its derivation. The case where the next treelet contains a matching trace with the current treelet can be seen as expanding the last added tree fragment. The case where the next treelet does not contain a matching trace can be seen as adding the first treelet of a new tree fragment to the derivation. In contrast to DOP, in EG the expansion of a derivation is conditioned on the derivation so far. This allows EG to be context sensitive.

6.2 Three types of tree fragments

This section displays my theoretical work of analyzing the types of tree fragments that can be constructed in the left-corner episodic grammar. The following three types of tree fragment structures have been found using the episodic shortest derivation parser. The next section will show how often these tree fragment types occur in the Brown corpus:

Contiguous tree fragment. A tree fragment in which all of its treelets are connected in a single derivation, i.e. they all form one derivation segment.

Discontiguous tree fragment. A tree fragment which contains two or more segments from a single derivation. This occurs when a parse tree contains a derivation segment which is interrupted by another derivation segment. Contiguous and discontiguous tree fragments are mutually exclusive.

Shift-centered tree fragment. When parsing with a left-corner-parser the shift operation creates a so called shift-loop in the derivation: the parse shifts from a treelet and projects its way back up to the same treelet. A shift-centered tree fragment is a derivation that includes an incomplete shift-loop. A shift-loop is incomplete if two treelets inside the loop do not share any connected traces. This results in a gap between the goal treelet and the last treelet in the shift-loop that still belongs to the same episode. The gap is represented by a node with the special symbol $\sim ?\sim$,

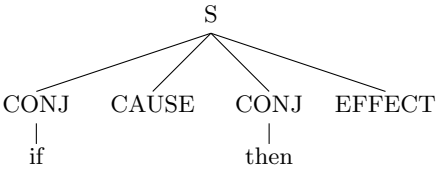
which connects these two treelets. A shift-centered tree fragment is either contiguous or discontiguous.

Figures 11, 12 and 13 show examples of these three types of tree fragments:

- Figure 11 shows a derivation of '*girl who dances likes tango*' that contains a contiguous tree fragment. The derivation of this tree fragment starts at $girl_{1-1}$ and stops at RC_{1-7}. Note that VI_{2-9} is part of this tree fragment, not because it is connected with the rest of the derivation, but because it is part of the treelet RC → WHO VI.

- Figure 12 shows a derivation that contains a discontiguous tree fragment. Just as with the contiguous tree fragment in figure 11, this discontiguous fragment starts at $girl_{1-1}$ and stops at RC_{1-7}. However, this time the derivation continues at RC_{1-10} and stops at S_{1-12}, thus making it discontiguous.

  Discontiguous fragments are able to represent long distance dependencies in a sentence[2]. Take for example the following discontiguous tree:



  This tree shows a long distance dependency between the words *if* and *then*. Another example of long distance dependency is the relation between the number of the subject and the verb in a sentence.

- Figure 13 shows a derivation that contains a shift-centered tree fragment. The derivation of this tree fragment starts at $girl_{1-1}$, projects towards NP_{1-4} and shifts downwards to who_{1-5}. After performing a shift the parser tries to find a path back to the goal treelet NP → NP RC, but instead stops at who_{1-5} before finishing the shift-loop. This creates a gap between the last parsed node and the left-most child of the goal treelet that has not yet been parsed (RC). This gap is indicated with the special symbol $\sim ? \sim$. The tree fragment shown in figure 13 is contiguous, since its derivation does not continue after ending at who_{1-5}.

Note that there is a difference between the tree fragments that are valid in the left-corner episodic grammar and in DOP. In DOP a tree fragment is valid when all of its nodes include all or none of their children. This means that a tree structure is only valid if they are connected in a top-down fashion. The episodic grammar, on the other hand, defines its tree fragments by its derivation, i.e. a set of treelets that are connected by their traces. A shift-centered tree fragment is valid in a left-corner episodic grammar, because its treelets are connected by means of a the shift-operation (see figure 13). Such a tree fragment, however, is not valid in DOP since there is no top-down connection.

All valid DOP tree fragments form a subset of the tree fragment that are valid in the left-corner episodic grammar (LC-EG). The connectivity in DOP between a node and its left-most child is equivalent to the project-operation in

[2]A context free grammar (CFG), for example, is not able to represent long distance dependencies, because it assumes that all of its rewrite-rules are context independent.

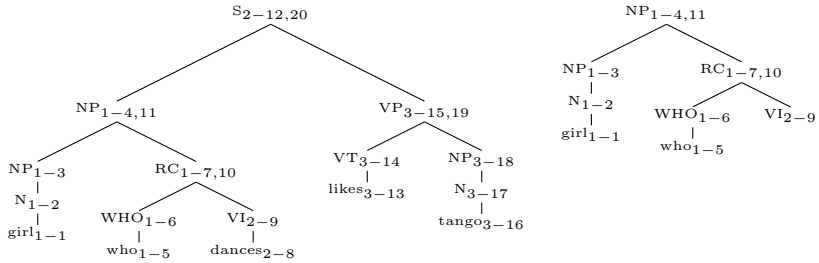


Figure 11: The left tree shows the derivation of the sentence '*girl who dances likes tango*'. The tree on the right shows one of its tree fragments that is contiguous. Traces are indicated in subscript.

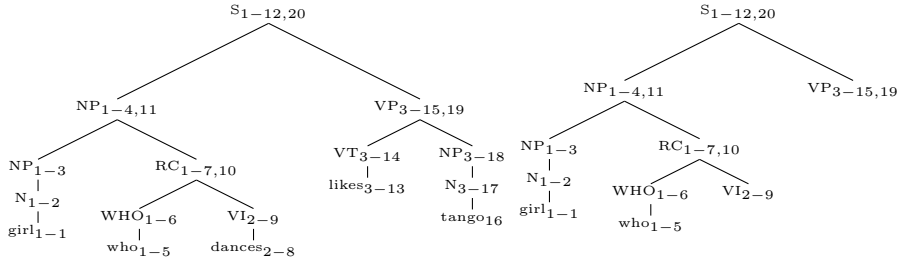


Figure 12: The left tree shows the derivation of the sentence '*girl who dances likes tango*'. The tree on the right shows one of its tree fragments that is discontiguous.

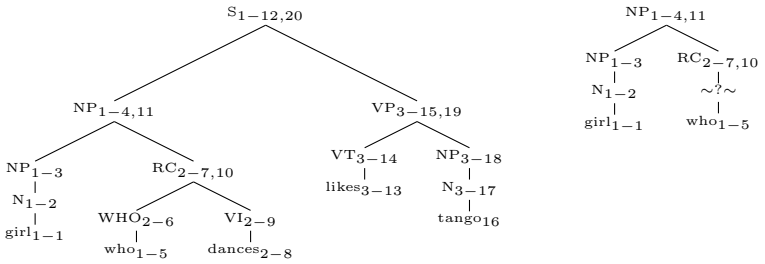


Figure 13: The left tree shows the derivation of the sentence '*girl who dances likes tango*'. The tree on the right shows one of its tree fragments that is shift-centered.

the LC-EG and the connectivity in DOP between a node and its non-left-most children is equivalent to the attach-operation in the LC-EG. The requirement in DOP that a node contains all or none of its children is also met in LC-EG since LC-EG uses treelets (a context-free-rules with traces) to represent a tree fragment.

7 Fragment measurements

This section will give a quantitative analysis of the fragments that are present in the Brown corpus [7] from the CHILDES database [10]. This corpus contains

sentences annotated with parse trees of three children. The data of two of these children, that of Adam and Eve, were used in the following experiments.

My supervisor, Gideon Borensztajn, provided a version of the Brown corpus that was annotated with traces from a left-corner episodic grammar that parsed according to the shortest derivation length. For each child the dataset was split into three parts that contain phrases from consecutive time periods. Table 1 shows the time periods for the children Adam and Eve. Using these datasets I was able to extract the tree fragments that made up the derivations of each sentence. Table 2 and 3 show the distribution of the tree fragments types, for Adam and Eve respectively. Note that since the contiguous and discontiguous tree fragments are mutually exclusive, the sum of both fraction always add up to one. Only a tiny fraction of the tree fragments found in the derivations seem to be shift-centered. This indicates that shift-centered fragments do not give a greater advantage over contiguous or discontiguous tree fragments when parsing a sentence.

Figures 14, 15 and 16 show some of the tree fragments of Adam in period 3 that were extracted from the corpus. Table 4 states the most frequent part of speech tags and grammatical relationships used in the Brown corpus. These, along with all the other tree fragments, are the first left-corner tree fragments that have ever been identified in an annotated corpus. Table 4 shows the syntactic categories that are used in the Brown corpus.

In a study conducted by Borensztajn et al. [6], DOP was used to identify the most likely primitive units that were used by children to produce the phrases in the Brown corpus. Tree fragments were extracted from the Brown corpus and used to calculate the most probable derivations (see equation 2). [3]. Their results confirm the *progressive abstraction* hypothesis: abstraction, defined as the ratio between the amount of non-terminal leaves and terminal leaves in a derivation, increases with age. This was shown to be true independently of sentence length. Their results are in accordance with the studies of Tomasello et al. which also show an increase of abstraction of the schemas children use to construct phrases (see section 2). I have attempted to reproduce their results using the left-corner episodic grammar. Figure 17 shows the abstraction ratio for different sentence lengths computed from Adam's dataset. Unfortunately, my results do not show an increase in abstraction, as was found by Borensztajn et al. The amount of non-terminal leaves seems to be roughly equal to the amount of terminal leaves over the three time periods. A possible explanation for this could be that the episodic grammar I used for this experiment, uses the shortest derivation to choose its derivation, while Borensztajn et al. choose the most probable derivation. A second experiment, using an episodic grammar that chooses the most probable derivation, would have to be conducted to find out if the left-corner episodic grammar is able to show an increase of abstraction over age.

8 Discussion

As discussed in section 5, the top-down and bottom-up parsers are not probable models of how the human parser processes a phrase. The former model is not data-driven while the latter suffers from a delay in interpretation. Even though

[3]The push-n-pull estimator was used to calculate the probability of the tree fragments.

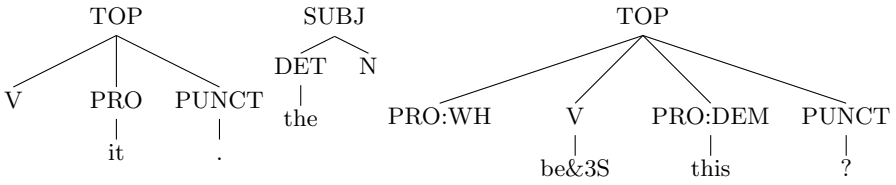


Figure 14: Some of the contiguous tree fragments extracted from Adam's dataset in period 3.

the left-corner-parser offers a more plausible parsing strategy, it does not mean that this is the same as the one used by the human parser. However, the lack of shift-centered fragments can be seen as evidence of how the human parser interprets a sentence. My results show that a word is almost never interpreted before it is attach to a goal treelet. I believe this reflects the uncertainty of choosing treelets in a derivation. If a word in not able to attach to a goal treelet, the parse would have to backtrack to a point before appending the goal treelet to the derivation. Only once the word is attached to goal treelet can the parse be more certain that the choices, that lead to adding the goal treelet to the derivation, have been correct. The more words that are attached to a goal treelet the more likely it is that this goal treelet will be used in the final derivation.

The lack of shift-centered fragments may also suggest that the human parser does not have a shift operation. It may be possible that it is not the shift-operation (caused by an incomplete treelet) that triggers the next word in a sentence to be parsed, but that each word starts a parse of its own whilst being guided by the previous treelets that need to be completed.

The results depicting the level of abstraction show that the shortest derivation episodic grammar is not able to confirm the progressive abstraction hypothesis. Further studies will need to be conducted in order to answer the question whether the left-corner episodic grammar is a suitable for modeling the way children parse language.

9 Preliminary research

It is still rather questionable whether the left-corner episodic parser correctly reflects how the brain represents and processes language. Nevertheless, this model offers a platform for developing techniques of unsupervised language acquisition. This section shows some preliminary work towards creating an unsupervised

	Age Range	
	Adam	Eve
Period1	2:3-2:11	1:6-1:9
Period2	2:11-3:6	1:9-2:0
Period3	3:6-4:5	2:0-2:3

Table 1: The begin and end of the children's age ranges are represented as *year:month*.

Adam	Period 1	Period 2	Period 3
No. sentences	7946	9307	7931
Average No. tree fragments	3,893405	4,350596	4,874039
Average depth	3,321042	3,796067	4,045139
Contiguous	0,482388	0,419342	0,441834
Discontiguous	0,517612	0,580658	0,558166
Shift-centered	0,024941	0,055861	0,058847

Table 2: Measurements over the three time periods of Adam.

Eve	Period 1	Period 2	Period 3
No. sentences	1859	2545	2446
Average No. tree fragments	2,949435	3,010216	3,198692
Average depth	3,173211	3,504912	3,690106
Contiguous	0,519521	0,476292	0,451110
Discontiguous	0,480479	0,523708	0,548890
Shift-centered	0,013535	0,025668	0,048890

Table 3: Measurements over the three time periods of Eve.

Parts of Speech	Category
N, N:PROP	Noun, Proper Noun
V, V:AUX	Verb, Auxiliary verb, including modals
Det, Det:NUM	Determiner (*the, a*), Number
ADJ, ADV	Adjective, Adverb
PRO, PRO:DEM	Pronoun, Demonstrative Pronoun (*this, that*), Interrogative
PRO:WH	Pronoun (*who, what*)
CONJ	Conjunction
INF	Infinitive marker (*to*)
PREP	Preposition
Grammatical relations	**Category**
TOP	Special category for the top node
SUBJ, OBJ	Subject Object
PRED	Predicative (He is not *sure*)
COMP, XCOMP	Clausal complements, finite and nonfinite
JCT	Adjunct (optional modifier of verb)
COORD	Coordination, dependents of the conjunction (*go* and *get it*)
AUX, NEG	Auxiliary and negation
LOC	Locative arguments of verbs (*in your truck*)

Table 4: Frequent Part of Speech and grammatical relations used in the Brown corpus.

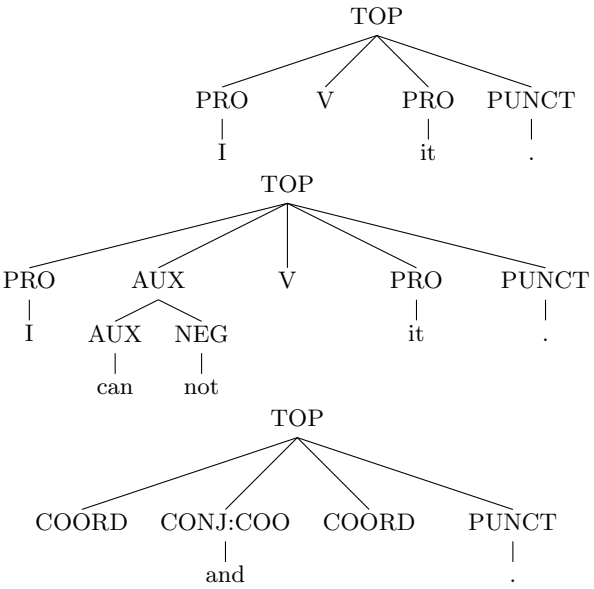


Figure 15: Some of the descontiguous tree fragments extracted from Adam's dataset in period 3.

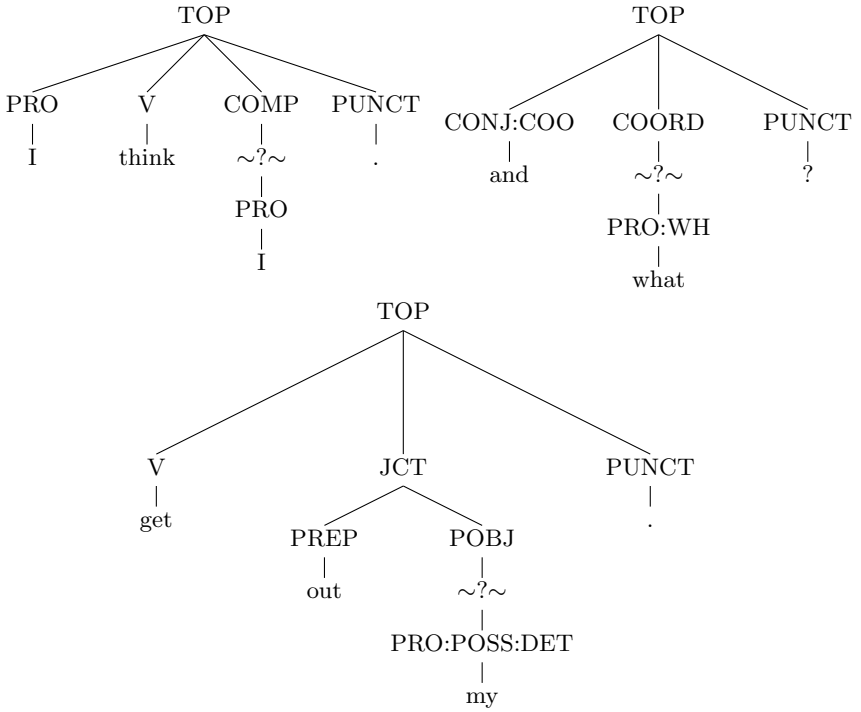


Figure 16: Some of the shift-centered tree fragments extracted from Adam's dataset in period 3.

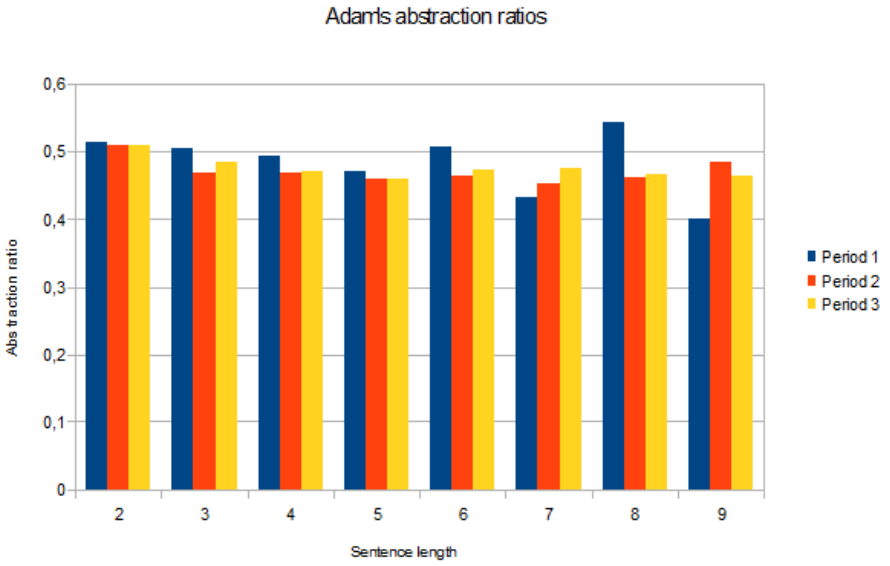


Figure 17: The abstraction ratio gives the ratio between the amount of non-terminal leaves and terminal leaves in the derivation of a sentence.

episodic grammar model.

An unsupervised language model does not need any sentences annotated with parse tree, but will have to learn these tree structures from scratch, the same way children learn their first natural language. Since episodic grammar is based on DOP, the logical step for me to take was to examine the unsupervised version of DOP (U-DOP). Whereas with DOP the challenge is to find the most probable derivation, U-DOP has the extra challenge of learning the correct tree fragments to use in its derivations. For each sentence, U-DOP starts off by generating all $C_n = \frac{(2n)!}{(n+1)!n!}$ possible binary trees [4], where $n-1$ is the length of the sentence. Then, as with DOP, the frequency of all subtrees are extracted and used to calculate the most probable derivation.

Even though it has been shown that U-DOP is able to construct correct tree structures for natural language (F-score of 78.5% on the Penn Tree Bank [4]), my findings indicate that U-DOP is not able to learn the correct tree structures of the simple grammar $a^n b^n$, which demonstrated center embedding. A correct binary tree of $a^n b^n$ may only contain the fragments X $\rightarrow$ a X, X $\rightarrow$ X b and X $\rightarrow$ a b (see figure 18). However, table 5 shows that the frequency of X $\rightarrow$ a a and X $\rightarrow$ b b is greater than the frequency of X $\rightarrow$ a b for $n \geq 3$. This means that if U-DOP is trained on sentences with $n \geq 3$, it is more likely to create an incorrect binary tree for $a^n b^n$ using the fragments X $\rightarrow$ a a and X $\rightarrow$ b b. These results show that U-DOP is not able to create the correcet tree structures for all context-free-grammars, however, its exact limitations remain unknown.

[4] C_n is the Catalan number

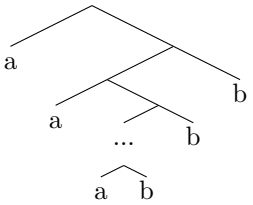


Figure 18: Binary tree of $a^n b^n$.

U-DOP($a^n b^n$)	n=1	n=2	n=3	n=4	n=5
No. X → a b	1	2	14	132	1430
No. X → a a	0	2	28	354	4433
No. X → b b	0	2	28	354	4433

Table 5: Frequencies of some of the tree structures created by U-DOP for the grammar $a^n b^n$.

10 Conclusion

In order to understand *how* a child is able to learn a natural language it is first important to understand *what* a child is learning. In this paper I explore the left-corner episodic grammar. This model offers a cognitively plausible model of how the hierarchical structure of language can be encoded in the brain using the semantic and episodic memory system; and it offers a parsing strategy that can more closely resemble the human parser than the top-down and bottom-up parser.

I have used this model to extract the contiguous, discontiguous en shift-centered tree fragments from the Brown corpus. This has given the first ever left-corner fragments from an annotated corpus. The results show that only a tiny fraction of the used tree fragments are shift-centered. I believe this reflects the uncertainty of choosing treelets in a derivation. A word that is not able to attach to a goal treelet would cause the parse to backtrack the derivation up to a point prior to adding this goal treelet. The more treelets that were able to attach to a goal treelet the more likely it is that the derivation so far has correctly interpreted the sentence.

Studies by Tomasello et al. have shown that children's grammar grow increasingly abstract over age. The tree fragments extracted by the left-corner episodic parser were not able to confirm the progressive abstraction hypothesis. This might be due to parsing the Brown corpus with the shortest derivation length. Further studies will need to be conducted in order to answer the question whether the left-corner episodic grammar is suitable for modeling the way children parse language.

References

[1] R. Bod. *Beyond Grammar: An experience-based theory of language.* CSLI Publications, Stanford, CA, 1998.

[2] R. Bod. Parsing with the shortest derivation. In *Proceedings of the 18th International Conference on Computational Linguistics (COLING 2000)*, -, 2000. Association for Computational Linguistics.

[3] R. Bod. An efficient implementation of a new dop model. In *Proceedings EACL'03*. 2003.

[4] R. Bod. Unsupervised parsing with u-dop. In *Proceedings of the 10th International Conference on Computational Natural Language Learning (CONLL-X)*, 2006.

[5] G. Borensztajn and W. Zuidema. Episodic grammar: a computational model of the interaction between episodic and semantic memory in language processing. In *Proceedings of the 33th Annual Conference of the Cognitive Science Society*, volume 9, 2011.

[6] G. Borensztajn, W. Zuidema, and R. Bod. Children's grammars grow more abstract with age - evidence from an automatic procedure for identifying the productive units of language. In *Proceedings of the 30th Annual Meeting of the Cognitive Science Society*, 2008.

[7] R. W. Brown. *A first language: The early stages*. Harvard University Press, Cambridge, MA, 1973.

[8] S. Crain and R. Thornton. Acquisition of syntax and semantics. In M. Traxler, editor, *Handbook of Psycholinguistics*. Elsevier, Oxford, 2005.

[9] M.W. Crocker. Mechanisms for sentence processing, 1998.

[10] B. MacWhinney. *The CHILDES project: tools for analyzing talk*. Lawrence Erlbaum Associates, Mahway, NJ, 2000.

[11] R. Scha. Taaltheorie en taaltechnologie: competence en performance. volume 11 of *7–22*, pages 409–440.

[12] R. Scha, R. Bod, and K. Sima'an. A memory-based model of syntactic analysis: data-oriented parsing. *Journal of experimenal and theoretical artificial intelligence*, pages 409–440, 1999.

[13] M. Smets. A u-dop approach to modeling language acquisition.

[14] M. Tomasello. *First verbs: a case study of early grammatical development*. Cambridge University Press, Cambridge, UK, 1992.

[15] M. Tomasello. First steps toward a usage-based theory of language acquisition. *Cognitive Linguistics*, 2000.

[16] M. Tomasello and P.J. Brooks. Young children's earliest transitive and intransitive constructions. *Cognitive Linguistics*, 9(4):379–395, 1998.

[17] E. Tulving. Episodic and semantic memory. In E. Tulving, editor, *Organization of memory*. Academic Press, New York, 1972.