*J. Child Lang.* **41** (2014), 176–199. 
doi:10.1017/S0305000912000694

# Computational evaluation of the Traceback Method

SHELI KOL
*Department of Computer Science, University of Haifa*

BRACHA NIR
*Department of Communication Disorders, University of Haifa*

AND

SHULY WINTNER
*Department of Computer Science, University of Haifa*

(*Received* 27 *August* 2011 – *Revised* 21 *March* 2012 – *Accepted* 11 *November* 2012 – *First published online* 24 *January* 2013)

ABSTRACT

Several models of language acquisition have emerged in recent years that rely on computational algorithms for simulation and evaluation. Computational models are formal and precise, and can thus provide mathematically well-motivated insights into the process of language acquisition. Such models are amenable to robust computational evaluation, using technology that was developed for Information Retrieval and Computational Linguistics. In this article we advocate the use of such technology for the evaluation of formal models of language acquisition. We focus on the Traceback Method, proposed in several recent studies as a model of early language acquisition, explaining some of the phenomena associated with children's ability to generalize previously heard utterances and generate novel ones. We present a rigorous computational evaluation that reveals some flaws in the method, and suggest directions for improving it.

## INTRODUCTION

Over the past two decades, an increasing number of studies in the domain of language acquisition have employed computational approaches. These studies are based on either symbolic models or, most prominently, on connectionist and probabilistic (Bayesian) models. For a critical review of

these approaches, see Alishahi (2010), Chater & Redington (1999), and Chater & Manning (2006).

In the domain of syntactic acquisition, computational studies examine the acquisition of a specific construction (e.g., simulating the developmental trajectories of finite and non-finite verb forms; Freudenthal, Pine & Gobet, 2006); they model induction of particular part of speech (PoS) categories (F. Chang, Lieven & Tomasello, 2008; Parisien, Fazly & Stevenson, 2008; Reali, Christiansen & Monaghan, 2003; Redington, Crater & Finch, 1998); or they examine several related constructions, for example, argument structure constructions (Alishahi & Stevenson, 2008; N. Chang, 2008). A few studies also attempt to account for the development of processing and production skills (e.g., Christiansen & MacDonald, 2009; F. Chang & Fitz, forthcoming).

Computational simulations that specifically implement definitions provided by cognitive models of language acquisition are rare. Lewis and Elman (2001) show that a neural network is able to generalize relative clauses from child-directed speech and correctly predict other complex syntactic structures (aux-questions) it has not encountered previously. Borensztajn, Zuidema, and Bod (2009) and Bod (2009b) show that abstraction in child speech, in the form of utterances that include schemas and slots, can be based on analogy and increases with age. Bannard, Lieven, and Tomasello (2009) examine the development of schemas and slots, comparing multiple possible context-free grammars induced from child speech. These studies follow on from a considerable body of experimental psycholinguistic research that has convincingly shown that children's early multiword utterances are restricted and non-novel, and constructed using rote-learned phrases, or LEXICALLY BASED PATTERNS. These patterns evolve at some point along development into less specific constructions that contain some level of abstraction (Lieven, Pine & Baldwin, 1997; MacWhinney, 1975; Peters, 1983; Tomasello, 2003). Such studies emphasize the major role of the child's ability to generalize structures – and particularly higher-level, constructional schemas (Dąbrowska, 2000) – from input data.

One of the main advantages of computational models is that they require formal, precise definitions of the assumptions underlying the model. Consequently, computational models can be rigorously EVALUATED: Their predictions can be put to test and their quality can be measured quantitatively. In this article we advocate such robust evaluation; we demonstrate it on one example of a psycholinguistically motivated study that provides explicit definitions based on a theoretical account of language acquisition.

The model we focus on is the TRACEBACK METHOD (henceforth, TBM) (Lieven, Behrens, Speares & Tomasello, 2003). The basic assumption of the

TBM follows the usage-based claim that language emerges as a result of various competing constraints which are all consistent with general cognitive abilities, and that language is first acquired in an item-based fashion (Bates & MacWhinney, 1987; MacWhinney, 1999; Tomasello, 2006). The TBM as a usage-based model follows the experimental studies noted above in relating children's constructions to input utterances. Specifically, the studies following this method aim to characterize what are termed NOVEL UTTERANCES, that is, productions that cannot be said to represent exact repetitions of previously heard utterances, and to show that even these productions can be closely related to specific constructions that emerge from adult input.

The TBM was introduced in several articles (Lieven *et al.*, 2003; Dąbrowska & Lieven, 2005; Bannard & Lieven, 2009; Lieven, Salomo & Tomasello, 2009; Vogt & Lieven, 2010), providing different definitions of the model. In the following section we introduce the model informally, trying to consolidate some of these differences. The result is a formulation of the model that can be computationally implemented. It is not clear to us whether such an implementation was actually carried out by the developers of the TBM; the various papers report results that may have been produced computationally or manually. Consequently, we reimplemented the model ourselves, using computational resources that were available to us (but which are not as precise as the resources used originally by Dąbrowska & Lieven, 2005, which are not publicly available). We provide a detailed description of our reimplementation. Then, we evaluate the model using standard measures of computational linguistics, and show that it vastly overgenerates.

The main contribution of this article, therefore, is that it highlights the need of robust evaluation of computational models of language acquisition. Such evaluation can shed light on suboptimal properties of the model being evaluated, and help improve it. While we only demonstrate this for a single model (albeit a very prominent one), we are confident that several works in the same domain (e.g., the MOSAIC model of Freudenthal, Pine & Gobet, 2007, 2009, 2010, or, more generally, the theory of Goldberg, 2006) can equally benefit from a better, more rigorous computational definitions and evaluation. See Bod (2009a) for a similar view.

## THE TRACEBACK METHOD

Throughout the various papers making use of the TBM, the method is presented as a finite set of ordered procedures, making it particularly appealing for a computational implementation. However, the set of operations varies from one paper to another, and no precise definitions (let alone an algorithm) are provided. In this section we describe the method informally,

trying to track some of the major changes introduced to it throughout the years. We present a formal, precise algorithm in the next section.

As noted above, the TBM maintains that children heavily rely on input in order to learn language. Children's productions are thus not manifestations of abstract rules, but exemplars of symbolic units, or linguistic constructions. As such, the multiword utterances produced by children contain both frozen, lexically specific, FRAMES, and generalized, open-ended SLOTS, into which some category of (typically content) words, the FILLER, can be integrated (Dąbrowska, 2000). Typical frame-and-slot constructions would be *There's a X*, *I want a Y*, and *Z it* (Dąbrowska & Lieven, 2005). In the following, we adopt a convention whereby frames are typeset in italics, and slots are specified by capital letters.

The first point addressed by the TBM is how to identify those parts in children's multiword utterances that are frozen and those that are schematic or constructed. According to Lieven *et al.* (1997), this procedure depends on the positional regularity of particular lexical items relative to all previously appearing utterances that contain the same words. In order to analyze children's productions, Lieven *et al.* (2003) introduce a procedure for matching children's utterances with utterances in the preceding corpus. In this procedure, a given utterance, the TARGET, is compared to all prior utterances that show lexical matching, on the one hand (based on the number of morphemes that are similarly sequenced), and variation in a particular position in the utterance, on the other.

In the simplest case, a target utterance can be matched against a preceding utterance such that exactly one morpheme distinguishes between the two. For example, if the target is *I got the butter* and a preceding utterance is *I got the door*, the procedure can yield the frame *I got the X*, where the *X* slot is filled by the fillers *butter* and *door*. In practice, though, a minimum number of preceding matching utterances is required in order for a frame to be well established; this number is determined to be two by Dąbrowska and Lieven (2005).

A more involved case is when a given target can be used to define two different, competing frames. Assume that the target is *Where's Annie's plate?* Preceding utterances include *Where's Annie's bottle?*, *Where's Annie's doll?*, *Where's Mummy's plate?*, etc. These give rise to two potential frames, namely *Where's Annie's X?* and *Where's X's plate?* In this case, the frame with the most CONSECUTIVE morphemes (here, *Where's Annie's X?*) is preferred. However, the criterion of consecutive morphemes is not always sufficient to break a tie between two competing frames; in such cases, frequency is used, and the frame with the most frequent instances is chosen.

This procedure is then used by Dąbrowska and Lieven (2005) for identifying what they term COMPONENT UNITS, or 'an expression which shares lexical material with the target and is attested at least twice in the main

corpus (excluding imitations and self-repeats)' (p. 447). Thus, SHARED MATERIAL can be 'any word or continuous string of words corresponding to a chunk of semantic structure … which occurs at least twice in the main corpus', and is thus defined as a FIXED PHRASE, while any 'string consisting of one or more fixed phrases and one or more slots of a specified kind … corresponding to a chunk of semantic structure' is considered a FRAME WITH SLOT (p. 447). Dąbrowska and Lieven (2005) then posit COMPONENT UNITS as symbolic constructions, allowing them to capture more abstract details of the internal organization of the children's utterances and to define a grammar that emerges from these data.

Dąbrowska and Lieven (2005) relate to particular symbolic units that represent semantic generalizations assumed to be available to the child (a THING, PROCESS, PROPERTY, LOCATION, DIRECTION, etc.). The use of symbolic units allows for tracing the 'semantic match between the items that create the slot and those that are inserted into it' (p. 444). These semantic units are then used to define two OPERATIONS that can be used to derive a target utterance from preceding ones.

The two operations defined by Dąbrowska and Lieven (2005) are JUXTAPOSITION and SUPERIMPOSITION. Juxtaposition is a 'linear composition of two units, one after another … in either order' (p. 442). In superimposition, one unit, the filler, 'elaborates a schematically specified subpart of another unit, the frame' (p. 443). Furthermore, 'the filler must match the properties specified in the frame'; so, if the frame calls for a THING, the filler must not be a DIRECTION.

For example, consider the target utterance *Where can he park now?* (after Dąbrowska and Lieven, 2005: 449). Attested frames include *Where can THING park?*, established from preceding utterances via the procedure described above. Since *he* is a component unit, whose semantic type is determined to be THING, it can be used as a filler and, via superimposition, yield the string *Where can he park?* In addition, *now* is also a component unit, which can be juxtaposed to this string, yielding exactly the target utterance.

These definitions and procedures have been tested, refined, and implemented in analyzing children's utterances in actual corpora. Dąbrowska and Lieven (2005) use this procedure to show that children's *wh*-questions can be traced back to the input presented to them (in other words, only child-directed speech (CDS) is used for deriving target utterances). They use a dense corpus, consisting of four developmental corpora for two English-speaking children, Annie and Brian, each recorded for six weeks at the ages of 2;0 and 3;0. Results show that approximately 90% of children's *wh*-questions can be generated by this model and, in line with previous studies, that 11–36% are direct repeats of utterances that already occurred in the main corpus (11% for Annie and 36% for Brian at age 2;0).

Moreover, at the age of 2;0, the majority of both children's utterances require only one operation for a successful derivation (55% for Brian and 66% for Annie). At age 3;0, a considerably higher proportion of utterances requiring two or (especially in the case of Annie) more operations is found, although many more of the children's *wh*-questions can still be derived by applying a single operation (25% for Annie and 43% for Brian).

More recent papers (Bannard & Lieven, 2009; Lieven *et al.*, 2009; Vogt & Lieven, 2010) implement the TBM on the full range of multiword utterances used by children. This version of the TBM was tested on data collected from a corpus of speech productions of the same two children analyzed by Dąbrowska and Lieven (2005) and of two additional English-speaking two-year-olds. It proved capable of tracing back between 83·1% and 95% of all child utterances, with around 25–40% of utterances constituting exact repetitions, and between 36% and 48% of utterances requiring just one operation for derivation. In Lieven *et al.* (2009), the quantitative criterion for schemas and fixed strings was set to one occurrence of the utterance in the prior data. This version of the model was tested on the child speech (CS) of the same four datasets as in Bannard and Lieven (2009), with highly consistent results, such that around 85% of the test data were traced back to the utterances said before. Vogt and Lieven (2010) repeat a similar procedure, this time using both CDS and CS as input, again with high success rates.

It thus seems that the TBM is consistently able to provide a constructivist account for the majority of the children's utterances on the basis of a lexically specific grammar that can be manipulated by a small number of general operations. Below, we reformulate the above informal description in terms of precise definitions, and spell out the algorithm which can be derived from these definitions. We then present the results of our reimplementation.

## REIMPLEMENTATION OF THE TBM

The advantage of the TBM is that it provides several definitions of key concepts in the studies surveyed in the previous section, including the input data, the units of analysis, and the procedures and operations that they rely on. The version of the TBM that we chose to implement is the one of Dąbrowska and Lieven (2005) (henceforth DL).

### *Definitions and materials*

First, the input data for the model are clearly and consistently defined as all multiword utterances (whether in CS, CDS, or both). The expected output is also defined: a list of utterances that can be shown to be inter-related

through a process of repetition and reuse. Note that the TBM does not induce a grammar, but rather proposes a process of increased schematization by which grammar emerges from data.

As noted, the TBM attempts to generate children's novel utterances using what they heard or said before. Not having access to the original corpus, we analyzed instead the online corpora of Brown (1973) and Suppes (1974), both available from the CHILDES website (MacWhinney, 2000). The Brown corpus contains transcribed longitudinal recordings of three American children, Adam (from 2;3 to 3;0, with a MLU of 1·55), Eve (from 1;6 to 2;3, with a MLU of 1·94) and Sarah (from 2;3 to 2;7, with a MLU of 1·8). From the Suppes corpus, we chose a subset of twenty-six files pertaining to Nina, an American girl, from age 1;11 to 2;5 (with a MLU of 2·9). These data are comparable to those used by DL, with MLU ranging across the period when multiword utterances emerge. However, these corpora are sparse, and as such may not be comparable to those used by studies of the TBM. Accordingly, we also analyzed two datasets of a British child, Thomas, recorded by Lieven *et al.* (2009). One set includes dense recordings of one hour five times a week, every week, for a period of a year (between the age of 2;0 and 3;2, MLU of 1·99, henceforth Thomas A) and the second includes recordings of five hours in one week of each month in the following year (from age 3;3 to around age 4;0, henceforth Thomas B). This makes our corpora more comparable to those reported on in the previous studies.

Similarly to the original method, each corpus was divided into two parts, TEST and MAIN. For each chronologically ordered file, we considered the last 10% of the child utterances in the file as the test corpus, and all earlier files, along with the first 90% of the adult and child utterances in the current file, as training utterances. While in computational linguistics it is sometimes common to perform (typically, 10-fold) cross-validation evaluations, the nature of our corpus, and in particular the importance of chronological order in child language development, dictate that a more random division of a corpus into train and test portions would be harmful. The size of each corpus (the number of MULTIWORD utterances and the number of word tokens) is detailed in Table 1.

The next set of definitions considers the units used in the analysis. The notions of frozen versus constructed utterances – or FIXED PHRASES versus FRAMES WITH SLOTS – remain the foundation of the analysis throughout the different studies reported on above. These, and the operations for deriving target utterances from prior data, lie at the heart of the analysis provided in the TBM studies. However, there is extensive variation in the number of operation types used in each version – from five (Lieven *et al.*, 2003), to two (Dąbrowska & Lieven, 2005), to a different set of two (Vogt & Lieven, 2010), and then to three operations (Bannard & Lieven, 2009). Given this

TABLE 1. *Size of the corpora*

| | Main corpus | | Test corpus | |
|---|---|---|---|---|
| Corpus | Utterances | Word tokens | Utterances | Word tokens |
| Eve | 19,536 | 85,350 | 224 | 875 |
| Adam | 20,443 | 75,213 | 792 | 3,166 |
| Sarah | 6,425 | 23,330 | 106 | 252 |
| Nina | 38,736 | 175,748 | 458 | 1,632 |
| Thomas-A | 25,776 | 132,836 | 357 | 1,269 |
| Thomas-B | 25,110 | 131,652 | 326 | 2,192 |

variability, we chose to implement the version of the model specified in Dąbrowska and Lieven (2005), as described below.

All data in the corpora used for the original TBM studies were manually annotated with semantic labels, following the assumption that children store pairings of phonological forms and semantic representations. Different examples for this level of annotation are given in the relevant literature, but only two are defined in detail, whereas an automated system requires a full set of such labels. In order to approximate the semantic labels of THING, PROCESS, ATTRIBUTE, LOCATION, DIRECTION, POSSESSOR, and UTTERANCE, while at the same time allowing for the emergence of other form–function pairings not explicitly specified in the TBM papers but that presumably formed part of the analysis, we added a distributional feature-matching component to the algorithm.

This component relies on Part-of-Speech (PoS) and dependency-relation tagging that are available as part of the CHILDES system: PoS tags for all corpora were produced by the MOR program (MacWhinney, 2000), a morphological analyzer, and dependency relations were derived by the GRASP program (Sagae, Davis, Lavie, MacWhinney & Wintner, 2010), a dependency parser for identification of grammatical relations in child-language transcripts. Each morpheme in the corpus was thus assigned two values: one specifying its lexico-syntactic category (noun, verb, adjective, preposition, etc.) and one specifying its relation with other elements in the utterance (subject, object, modifier, locative, etc.). While the codes are generated automatically, they are quite reliable. For the Eve corpus, all codes were manually verified, so they can be assumed to be 100% correct. Accuracy of the annotation on other corpora is lower, but even so, the morphological tags are accurate in over 97% of the tokens, whereas Sagae *et al.* (2010) report syntactic error rates of only 5·4% (on adult utterances) to 7·2% (on child utterances). The main grammatical relations, such as subject and object, are accurate in over 94% of the cases.

We utilize these two levels of annotation in a way that complies with the procedure specified in Lieven *et al.* (2003: 349). Whenever the procedure

calls for two phrases to have matching semantic labels, our algorithm approximates the generalizations that underlie the categories assumed by the manually tagged semantic labels by requiring that the two phrases have identical PoS and dependency tags. This is in fact a more restrictive criterion than the one used by DL, since sometimes phrases can have the same semantic label even if their syntactic function is not identical; but we prefer to err on the side of being more restricting when we address issues of overgeneration.

As an example, consider the three utterances *Can we fix it?*, *Can we dig it?*, and *Can we lose it?* They all share the same PoS tag sequence, namely *aux pro v pro*; furthermore, they are all associated with the same syntactic structure: the verb is the root, and its three dependents are the first word *Can* (annotated as AUX by GRASP), the second word *we* (SUBJ) and the final word *it* (OBJ). Such utterances reflect the construction *Can we PROCESS it?*, where PROCESS is the single slot. Note that our algorithm does NOT generate the label PROCESS explicitly; it is only assumed, by the combination of the PoS tag *v* and the grammatical relation *root*. But we continue to use such labels below for brevity.

More variation in the object position, for example, reflected by the examples *Can we fix it?* and *Can we help you?* (again with the same PoS and syntactic structure as above) yields the more abstract construction *Can we PROCESS THING?*; and similarly, more variation in the subject position, demonstrated by *Can we fix it?*, *Can we help you?*, and *Can I do it?*, yields the even more abstract construction *Can THING PROCESS THING?* The labels of the slots that are determined in this way are then used when potential operations are considered, in line with the specification of DL.

### *The TBM algorithm*

We define the various stages of the TBM operation as an algorithm. In the following, meta-variables $S$, $T$, $u$, $w$, with or without subscripts, range over non-empty strings of words; meta-variables $\alpha$, $\beta$, $\gamma$, with or without subscripts, range over possibly empty strings of words. We write *match*($w$, $u$) when $w$ and $u$ are of the same length, and their (PoS and syntactic) annotations are identical. The input is a target utterance, and the output is a derivation ('traceback') of this utterance using superimposition and juxtaposition operations, if such a derivation is possible.

Given an annotated training corpus and a target utterance $T$, viewed as a string of words, the algorithm operates as follows:

1. Identify in the training material all COMPONENT UNITS with respect to the given target utterance, $T$. A non-empty string $S$ is a component unit

of $T$ if $T=\alpha S\beta$ for some possibly empty strings $\alpha$, $\beta$, and $S$ occurs at least twice in the training corpus. For each component unit $S$, store also all utterances in the training corpus of which $S$ is a substring.

2. If $T$ is available to the child as a component unit (i.e., it is a FIXED STRING), exit: the derivation is defined by this unit. Formally, if a component unit $S=T$ exists, return $S$ with 0 operations.
3. Otherwise, for each component unit $S$ of $T$, and for each training corpus utterance $u$ of which $S$ is a substring, find the longest match between $T$ and $u$. Formally,
   - If $T=\alpha Sw$ and $u=\gamma_l Sw'\gamma_r$ and *match*($w$, $w'$) then $S$ is a FRAME, $w$ is a FILLER and the operation is defined as *Superimpose*($S$, $w$).
   - If $T=wS\beta$ and $u=\gamma_l w'S\gamma_r$ and *match*($w$, $w'$) then $S$ is a FRAME, $w$ is a FILLER and the operation is defined as *Superimpose*($w$, $S$).
   - If $T=\alpha SwS'\beta$ and $u=\gamma_l Sw'S'\gamma_r$ and *match*($w$, $w'$) then ($S$, $S'$) is a FRAME, $w$ is a FILLER and the operation is defined as *Superimpose*($S$, $w$, $S'$).

   From all utterances in the training corpus, choose the one for which the frame length $|S|+|S'|$ is maximal. If more than one exists, pick one arbitrarily. Call recursively with $w$ as the target.
4. If no such frame is found, let $S$ to be a longest substring of $T$ that is available to the child as a component unit. Formally, let $S$ be a longest string such that $T=\alpha S\beta$. If no such string exists, fail.
5. If more words exist in $T$, call the algorithm recursively for the remaining utterance. Formally, in all the cases of Steps 3 and 4 above, $T=\alpha S\beta$ or $T=\alpha SwS'\beta$; call recursively with $\alpha$ if it is non-empty, and with $\beta$ if it is non-empty, and define the result as *Juxtapose*($\alpha$, $S$) (if $\beta$ is empty) or *Juxtapose*($S$, $\beta$) (if $\alpha$ is empty) or *Juxtapose*($\alpha$, *Juxtapose*($S$, $\beta$)) (if both are non-empty).

   If $T$ is accounted for in its entirety, report the number and types of the operations used to derive it.

This algorithm thus implements the 'rules' specified by DL (and reiterated in Bannard and Lieven, 2009), guaranteeing that derivations use the minimum number of operations. The recursive nature of the algorithm induces a hierarchical structure on utterances. Component units can function as fillers, in which case they have to be accounted for (traced back) themselves.

Example (1), taken from Eve's last data file (age 2;3) and including the MOR and GRASP tags, illustrates the process of deriving target utterances with the implementation of DL's algorithm.

(1)
```
*EVE: you  can help me
%mor: pro  aux v    pro
%gra: SUBJ AUX ROOT OBJ
```

```
*MOT: you  can  write one
%mor: pro  aux  v     pro
%gra: SUBJ AUX  ROOT  OBJ

*MOT: you  can  write me  a   lady on   this page
%mor: pro  aux  v     pro det n    prep pro  n
%gra: SUBJ AUX  ROOT  OBJ DET  OBJ2 JCT  DET  POBJ

*MOT: you  can  tell him now
%mor: pro  aux  v    pro adv
%gra: SUBJ AUX  ROOT OBJ JCT
```

The first utterance, *you can help me*, is the target. In order to trace it back, the algorithm first searches for the longest component units. Since the target utterance itself is not found in the data, its parts are the next candidates. Both *you can* and *help me* are found at least twice in the training corpus, and thus qualify as component units. Next, the algorithm searches for frames and slots. Given the other strings in Example 1, *you can* yields a potential frame: *you can PROCESS THING*, where both PROCESS and THING are slots of the frame that should be filled by fillers identified as belonging to the PROCESS and THING categories, respectively. This frame with slots is based on the shared material stemming from *you can make one*, *you can write me*, *you can tell him*. However, as in the examples provided by DL, since *help me* is available as a component unit, only a single operation is required to derive the target utterance, by superimposing the fixed phrase over the slot in *you can PROCESS*. Note that this is the preferred choice since it involves a smaller number of operations. No further derivation is required.

As another illustration of how the algorithm works, example (2) presents an utterance that is derived by two operations of superimposition.

(2)
```
*EVE: she  just     sitting there
%mor: pro  adv:int  part    adv:loc
%gra: SUBJ JCT      ROOT    JCT
```

The target utterance does not occur as a fixed string; the strings *she just sitting* and *just sitting there* are not found as fixed strings either. Of the other possible combinations, *sitting there* is found as a fixed-string component unit (appearing twice in the training corpus). This yields the following frame with slots: *THING just PROCESS*, into which the pronoun *she* and the phrase *sitting there* are superimposed.

We now present analyses of the various datasets chosen for this study, based on the implementation of this algorithm.

TABLE 2. *Re-implementation results: for each corpus, the number and ratio of successfully derived utterances*; *of those, the number and ratio of utterances derived using exact matches* ('*Fixed*'), *using any of the two operations* ('*Superimpose*' *and* '*Juxtapose*'); *and of the utterances derived by some operation, the number and ratio derived using only one or two operations*

| | | Derived | | Fixed | | Superimpose | | Juxtapose | | 1 OP | | 2 OP | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Corpus | Test | # | % | # | % | # | % | # | % | # | % | # | % |
| Eve | 224 | 155 | 69 | 37 | 24 | 87 | 56 | 32 | 21 | 95 | 81 | 11 | 9 |
| Adam | 792 | 675 | 85 | 183 | 27 | 312 | 46 | 185 | 27 | 362 | 74 | 70 | 14 |
| Sarah | 106 | 94 | 89 | 40 | 43 | 45 | 48 | 9 | 10 | 54 | 100 | 0 | 0 |
| Nina | 458 | 401 | 88 | 119 | 30 | 217 | 54 | 66 | 17 | 230 | 82 | 27 | 10 |
| Thomas-A | 357 | 246 | 69 | 106 | 43 | 136 | 55 | 8 | 3 | 127 | 91 | 13 | 9 |
| Thomas-B | 436 | 260 | 60 | 101 | 39 | 150 | 58 | 12 | 5 | 143 | 90 | 15 | 9 |

## RESULTS

Table 2 presents the results of implementing the procedures described in the previous section, with information about the number of utterances in the test corpus and the number of utterances that the algorithm successfully derives. Of those derived utterances, the table also reports how many utterances were derived by FIXED PHRASE (i.e., an exact match was found in the main corpus), how many utterances were derived by superimposition or juxtaposition, and how many by a single operation or by two operations.

The results of our reimplementation over six different corpora show that between 60% and 89% of the children's utterances in the test corpus can be derived using this algorithm, with most of the derivations based on one operation only. In other words, the bulk of the target data are generated by the computer program implementing the TBM. These results also show that out of the successfully derived utterances, between 24% and 43% were exact repetitions of previously heard utterances. It is noteworthy that a significant portion of all test utterances were exact repetitions of utterances that were previously heard or produced by the child, even in a relatively sparse sample. These results are compatible with those obtained in all the TBM studies mentioned above, and especially in Bannard and Lieven (2009), who use this method to trace back all the utterances in the child test corpus and report between 25% and 40% of exact repetitions. Our findings can thus be taken as supporting evidence to the claim that children in fact learn chunks from what they hear.

Our results are nonetheless significantly lower than those reported in the original TBM studies. For example, Bannard and Lieven (2009) were able to generate as many as 95% of all child utterances. One main reason for this discrepancy could be that our analysis is carried out on a much sparser

corpus, which makes the induction task more difficult for the system. However, the results for the dense Thomas corpus are highly comparable to those obtained for both the more sparse Thomas corpus and the various other corpora that were used in reimplementing the TBM. This poses a question with respect to the view that relying on a dense corpus affects the variability of the syntactic structures in use (Demuth, 2008).

The source of the difference between the results of the two implementations may also be attributed to the fact that our 'semantic' approximations of slot types is done automatically rather than manually, by means of a combination of morphological and syntactic annotation of the data. This in itself might have introduced another level of noise. Yet, again, our results do by and large correspond to the original TBM findings. This conclusion is supported by particular comparisons provided in the next two sections: types of operations required for derivation, and results when relying only on child-directed speech or child speech as the input for the algorithm.

### *Types of operations*

As shown by the results presented in Table 2, the rate of superimposition and juxtaposition operations used in most of our test corpora is considerably different from the results reported on in the TBM studies: in our analysis, superimposition accounts for no more than 58% of the utterances (in the dense Thomas corpus) while juxtapositions are employed in as many as 27% of the utterances (in the Adam corpus). On the other hand, type of operation seems to be corpus-dependent. Thus, only 10% of the Sarah test corpus was derived by use of juxtaposition, and, for both of the Thomas corpora, percentages are as low as in the original studies: 3% and 5%, respectively (consistently with the TBM reported results of never exceeding more than 5%; Lieven *et al.*, 2009). As to the number of operations, similarly to the TBM studies a relatively small number of utterances required more than one operation. Importantly, very few utterances required both superimposition and juxtaposition for derivation, such that most instances of multiple operations were of more than one superimposition. This result lends support to the major role played by this type of operation in analyzing the data.

### *Implementation based on CDS or CS only*

Different conditions were tested by the algorithm in the various TBM studies. Dąbrowska and Lieven (2005) examine only question constructions, whereas Bannard and Lieven (2009) and Vogt and Lieven (2010) use all child utterances for their implementation; and while these studies use both CDS and CS as the training corpus, Lieven *et al.* (2009) rely only on CS.

TABLE 3. *Results for training data as CDS or CS only*

| | | Derived | | Fixed | | Superimpose | | Juxtapose | |
|---|---|---|---|---|---|---|---|---|---|
| Corpus | size | # | % | # | % | # | % | # | % |
| Eve-CDS | 224 | 155 | 69·2 | 37 | 23·9 | 87 | 56·1 | 32 | 20·6 |
| Eve-CS | | 155 | 65·6 | 33 | 22·5 | 71 | 48·2 | 44 | 29·9 |
| Adam-CDS | 792 | 632 | 79·8 | 131 | 20·7 | 277 | 43·8 | 234 | 37·1 |
| Adam-CS | | 649 | 81·9 | 160 | 24·7 | 294 | 45·3 | 202 | 31·1 |
| Sarah-CDS | 106 | 91 | 85·9 | 28 | 30·8 | 51 | 56·1 | 12 | 13·2 |
| Sarah-CS | | 79 | 74·5 | 32 | 40·5 | 33 | 41·8 | 14 | 17·7 |
| Nina-CDS | 458 | 408 | 89·1 | 103 | 25·2 | 227 | 55·6 | 79 | 19·4 |
| Nina-CS | | 395 | 86·2 | 119 | 30·1 | 196 | 49·6 | 84 | 21·3 |
| Thomas-A-CDS | 357 | 202 | 56·6 | 44 | 21·8 | 151 | 74·8 | 19 | 9·4 |
| Thomas-A-CS | | 246 | 68·9 | 106 | 43·1 | 136 | 55·3 | 8 | 3·3 |
| Thomas-B-CDS | 436 | 233 | 53·4 | 78 | 33·5 | 143 | 61·4 | 14 | 6·0 |
| Thomas-B-CS | | 242 | 55·5 | 103 | 42·6 | 131 | 54·1 | 11 | 4·6 |

Both types of input yielded successful tracebacks. The results of our current application comparing derivations based on CDS versus CS are presented in Table 3.

The data in Table 3 show that the TBM algorithm works just as well both when the training data is only CDS and when it is only CS. That is, the percentages of derived utterances and of superimposition and juxtaposition are highly comparable both with each other and with the general (all-inclusive) results. This supports the finding of Lieven *et al.* (2009), according to which children's utterances are closely related to what they themselves said before, but it also suggests that the children's corpus is very much related to the adult input, even if less than 90% of the data can be derived.

It seems, then, that our reimplementation of the TBM algorithm is successful: even though we observe a somewhat lower percentage of derivations, tracing back fixed strings and applying superimposition and juxtaposition operations account for a large amount of the children's utterances. But is this evaluation sufficient? The next section addresses this question.

## EVALUATION

The preceding analysis reveals the TBM as a model which is explicitly and formally stipulated and cognitively well motivated. This makes it possible to test and corroborate its results. On the other hand, closer examination of its underlying definitions revealed two issues as insufficiently defined across the various versions of the model. First, the characterization of the various slots used in these studies is not extensive, and the list of possible types of

slots is not comprehensive. Second, the constraints on the two operations, superimposition and most notably juxtaposition, are not sufficiently well defined. Although we were able to implement the model, we could only approximate the suggested slots. This has potentially contributed to the lower percentages of derivations obtained for the test data. The lack of explicit constraints on the operations could also mean that the generative power of our algorithm is greater than that of the manually obtained analyses, especially in those datasets that showed a much higher percentage of juxtaposed ordering of component units. Below, we examine the implications of these conditions on the current TBM application by computationally evaluating its overgeneration capacities.

Several factors make the evaluation of language-learning systems difficult (Zaanen & Geertzen, 2008), and two of them stand out in the present context. First, the training data provided to the system (i.e., the corpus used for induction) are usually limited. This is especially true when child data are concerned, such that even with high-density corpora it is assumed that the corpus reflects less than 10% of the utterances the child was exposed to during a very short period (see Rowland, Fletcher & Freudenthal, 2008). It is thus hard to evaluate the quality of the generalizations performed by the system. Second, and more crucially, while it is relatively easy to measure the proportion of the target utterances that the system properly generated, it is much harder to assess the proportion of the utterances generated by the model that are indeed grammatical. Especially in the context of child language, it is always hard to determine what constitutes an ill-formed utterance. We now elaborate on this difficulty.

In the computational linguistics community, similar tasks are standardly evaluated using two measures adopted from Information Retrieval: RECALL and PRECISION. The task is defined as GRAMMAR INDUCTION: given a sequence of (training) utterances, a model is supposed to generalize them, typically by representing them in a compact form, as a GRAMMAR. Then, this grammar can be evaluated by applying it to a set of other utterances, the TEST set. Informally, RECALL measures the ability of the grammar to account for new utterances. It is the proportion of the test utterances that the grammar can properly generate. PRECISION, on the other hand, measures the extent to which grammar-generated strings are observed in real life; it is the proportion of the set of generated utterances that are indeed correct. The precision thus goes down when the grammar overgenerates.

More formally, assume that a test set consists of both positive (i.e., grammatical) and negative (i.e., ungrammatical) examples. The grammar is required to determine the grammaticality of each of these examples. Let TP be the number of positive examples the grammar correctly judged as grammatical; TN the number of negative examples the grammar correctly judged as ungrammatical; FP the number of ungrammatical examples the

grammar mistakenly judged as grammatical; and FN the number of positive examples the grammar mistakenly judged as ungrammatical. Then the precision is $p = \text{TP}/(\text{TP} + \text{FP})$, and the recall is $r = \text{TP}/(\text{TP} + \text{FN})$.

It is important to note that a clear trade-off exists between the two measures, such that it is always possible to improve one measure at the expense of the other. In the extreme case, consider a grammar that generates the empty language: such a grammar will of course have zero recall, but an impeccable precision (since it never overgenerates). At the other extreme, a grammar that generates everything, all possible sequences of words, will have perfect recall but an extremely low precision.

The recall of grammar induction algorithms is easy to compute: one has to design an appropriate test set and measure the ability of the algorithm to cover it. It is much harder to evaluate precision, however, because it is unclear what to use as a test set: the set of utterances that can be generated by a grammar induction algorithm is typically infinite, and in any case huge. To know that an algorithm overgenerates, what is needed is a set of UNGRAMMATICAL utterances, or utterances which the grammar was not supposed to have learned. Such sets are usually unavailable.

This difficulty is addressed in various ways. One approach uses an alternative model, which is assumed to be correct (Berant, Gross, Mussel, Sandbank & Edelman, 2007). Here, strings generated by the evaluated model are tested on an alternative model, which can determine whether the generated strings are indeed grammatical. However, it is usually impossible to obtain alternative models that can learn exactly the same language as the one induced by the evaluated model. Another approach uses human judgments (Solan, Horn, Ruppin & Edelman, 2005; Brodsky, Waterfall & Edelman, 2007): strings generated by the grammar are judged by human evaluators. Such judgments are often subjective and unreliable, especially when child-language data are concerned. Bannard *et al.* (2009) use a measure of PERPLEXITY for evaluating the quality of the grammars they induce. However, perplexity evaluation requires very long strings (it is an approximation of an infinitely long sequence of words), and it is not clear how it can be applied to a situation where most sequences are shorter than ten words.

Finally, F. Chang *et al.* (2008) propose a specific measure for evaluating the quality of syntax learners, SENTENCE PREDICTION ACCURACY (SPA). Given a test sentence, its words are viewed as a set (bag of words), and the grammar is requested to determine, incrementally, the most plausible ordering of the words in the set. The grammar constructs an incrementally longer sequence of words from the bag by selecting, at every iteration, the one word from the remaining words in the bag whose likelihood to follow the currently constructed sequence is highest. The end result must be identical to the original test sentence. One disadvantage of this method is

that it is extremely strict, in the sense that a mistake in the placement of a single word renders the entire utterance unaccounted for. In addition, this method assumes that the grammar has a way to determine which of several word sequences is more likely; this makes the method inapplicable in our present setting, where no probabilities are involved.

In lieu of an accepted method for evaluating the precision of language-learning algorithms, we suggest a method to assess the level of over-generation of a grammar learner that is inspired by SPA (F. Chang *et al.*, 2008) but which does not require probabilities and can be easily applied to our scenario. The idea is to test the induced grammar on strings that are less likely to be grammatical than the actual utterances in the corpus. Since it is generally very hard to determine what constitutes an ungrammatical string, especially in the context of child language, we capitalize on the observation that English is a relatively fixed-word-order language, and assume, like F. Chang *et al.* (2008), that an actual utterance is typically 'more grammatical' than any other ordering of its words. In other words, we assume that if we test a grammar on actual utterances, it should succeed in generating more of them than it would had we tested it on the same utterance, but where each utterances had its words reordered.

We thus create two corresponding TEST corpora for our existing child corpus, containing all the original utterances in the test corpus of length 2 or more. In one corpus, REVERSED, each utterance is listed in reverse word order; in the other corpus, RANDOM, the words of the original test utterance are ordered randomly. For example, if the original test utterance is *you can do it the other way*, then its reverse counterpart is *way other the it do can you*, whereas a random instance could be *other you can way it do the*. We then re-run the algorithm, training it, as in the previous section, on CDS and CS together, but testing it on the reversed and random utterances. Crucially, the algorithm is still trained on the ORIGINAL utterances, so it is expected to learn the actual language reflected in the corpus; but the learner is now evaluated on its ability to generate correctly not only actual utterance, but additionally also reversed and random utterances.

Table 4 compares the original results for each corpus with the derivation of reverse and randomly ordered utterances. As can be seen in the table, the results are highly comparable. For example, on the Eve corpus, the algorithm can derive 69% of the test utterances, but also 68% of the reverse utterances, and 65% of the randomly ordered ones. The same pattern recurs for the Adam corpus, where the algorithm derives 85% of the test utterances, but also 83% of the reversed and random-ordered ones. The other corpora reveal a very similar pattern. This indicates a serious overgeneration problem: while our implementation of the TBM is able to learn a significant portion of all child utterances even for a non-dense corpus, this very ability may be interpreted as evidence of the system's

TABLE 4. *Evaluation of the TBM*

| | | Derived | | Fixed | | Superimpose | | Juxtapose | | 1 OP | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| Corpus | size | # | % | # | % | # | % | # | % | # | % |
| Eve | 224 | 155 | 69 | 37 | 24 | 87 | 56 | 32 | 21 | 95 | 74 |
| (reversed) | | 153 | 68 | 5 | 3 | 64 | 42 | 84 | 55 | 93 | 63 |
| (random) | | 146 | 65 | 29 | 20 | 68 | 47 | 49 | 34 | 78 | 67 |
| Adam | 792 | 675 | 85 | 183 | 27 | 312 | 46 | 185 | 27 | 362 | 83 |
| (reversed) | | 659 | 83 | 43 | 7 | 226 | 34 | 403 | 61 | 335 | 54 |
| (random) | | 657 | 83 | 138 | 21 | 271 | 41 | 260 | 40 | 313 | 60 |
| Sarah | 106 | 94 | 89 | 40 | 43 | 45 | 48 | 9 | 10 | 54 | 100 |
| (reversed) | | 94 | 89 | 8 | 9 | 63 | 67 | 23 | 25 | 82 | 95 |
| (random) | | 91 | 86 | 8 | 9 | 73 | 80 | 12 | 213 | 80 | 98 |
| Nina | 458 | 401 | 88 | 119 | 30 | 217 | 54 | 66 | 17 | 230 | 81 |
| (reversed) | | 389 | 85 | 22 | 6 | 143 | 37 | 227 | 58 | 241 | 63 |
| (random) | | 400 | 87 | 71 | 18 | 132 | 33 | 198 | 50 | 248 | 75 |
| Thomas-A | 357 | 246 | 69 | 106 | 43 | 136 | 55 | 8 | 3 | 127 | 90 |
| (reversed) | | 204 | 57 | 48 | 24 | 150 | 74 | 15 | 7 | 123 | 79 |
| (random) | | 212 | 59 | 81 | 38 | 131 | 62 | 3 | 1 | 114 | 5 |
| Thomas-B | 436 | 260 | 60 | 101 | 39 | 150 | 58 | 12 | 5 | 143 | 90 |
| (reversed) | | 213 | 49 | 31 | 15 | 177 | 83 | 11 | 5 | 136 | 43 |
| (random) | | 219 | 50 | 88 | 40 | 129 | 59 | 6 | 3 | 106 | 79 |

overgenerative power. Indeed, it can also derive 49–89% of the reversed and randomly ordered utterances (of length 2 or more), that is, of what are most likely ungrammatical structures.

As examples of successful derivations of reversed utterances, consider the (reversed) utterance *Boston to go*, which is generated by juxtaposing *Boston* with the component *to go*. A single juxtaposition operation also suffices to generate the reversed utterances *Gloria you're* and *more no*. Superimposition is used to generate *it eat* as an instance of the schema *it PROCESS*, where *eat* fills the PROCESS role (the original utterance, of course, is *eat it*), and *drink want* is derived from applying superimposition of *THING want*. Also, *crying her* is derived from *PROCESS her*, a construction that also derives *see her* and *put her*. And the use of one superimposition and one juxtaposition generated the reversed utterance *more buy to have* as an instance of the schema *PROCESS to have*.

Such examples abound, and include also longer sequences. Consider the reversed target utterance *one another write now*. The presence of *oh I see what you're doing now* and *what are you going to do now* in the training material gives rise to the frame *PROCESS now*. This is then filled by *write*, yielding *write now* with one superimposition operation. Two juxtapositions then generate the target. Even if one were to limit the number of juxtaposition operations to at most one per derivation, clearly ungrammatical strings could be derived. Consider the reversed target *look me want you do?* The sequence *let me help you* occurs twice in the training material, and thus

gives rise to the frame *PROCESS me PROCESS you*. The two slots of this frame are filled by *look* and *want*, and then a single juxtaposition suffices to add the final *do*.

Of course, not ALL strings in the corpus can be derived. Several strings fail because of lexical omissions: they involve words that occur fewer than twice in the training material. Examples include *yeah I need rifle*, with *rifle* occurring only once in the training set; or *do you want me take rubberband off?*, with *rubberband* occurring only once. Other strings fail to be derived on different grounds, most notably type mismatches between a slot and its candidate filler. For example, the target utterance *the rich live* could not be traced back; the training material does include two instances of *the rich*, namely *the rich people live in that castle?* and *the rich people?*, which give rise to the frame *the rich THING*. However, in the target, *live* is annotated as a verb, and hence can only fill a PROCESS slot in a frame, not a THING slot. Similarly, the training material includes both *listen about cowboy on a big train coming toot toot* and *a nice train?*, from which the frame *a ATTRIBUTE train* is generated. This frame is a candidate for deriving the target utterance *a monkey train*, but since *monkey* is a noun, its label is a THING and it fails to fill the slot of the frame.

The results presented in Table 4 clearly indicate that the juxtaposition operation needs to be more constrained. While superimposition is constrained by the type of the slot, juxtaposition is not, and, in particular, it allows either order of combination. The analysis of reverse utterances yielded twice as many uses of juxtaposition, accounting for more than half of all derivations for the Eve, Adam, and Nina corpora. Yet even given the non-constrained nature of this operation, it still failed to account for the entire corpus, indicating that what is involved here is more than stringing one word after the other. But, as it stands, the algorithm does allow more of this operation than is desirable. As noted above, several attempts were made to constrain this operation, for example in restricting the application of juxtaposition (called ADD in that version of the method) by suggesting that it is 'only allowed if the component unit could, in principle, go at either end of the utterance' (Bannard & Lieven, 2009), and Vogt and Lieven (2010) suggest that only specific items (such as vocatives) can participate in this operation. However, it is unclear how these restrictions are determined, or even how the child could know them.

Another issue that contributes to the overgeneration of the algorithm is the unlimited number of operations. True, DL postulate a 'minimal number of operations' requirement, but they specify no upper limit. Our analysis of the reverse corpus requires a larger number of multiple juxtaposition operations in order to allow derivation, since no fixed strings were found in the corpus. Conceivably, such a situation could emerge for

non-reverse corpora. In order to avoid recursion of this operation, some constraint must be suggested.

Finally, and importantly, the TBM does not consider the frequency with which utterances are presented to the learner, and so does not take into account the effect of recurring strings on the entrenchment of linguistic structures (Bybee, 1995, 2006). A model that is more sensitive to frequency effects is likely to better fit the data, both in its regular and reverse versions.

## CONCLUSION

The purpose of this article was to computationally evaluate a given psycholinguistic model of the way children acquire language and to contribute to the domain of cognitive linguistics by both corroborating and criticizing usage-based assumptions via computational and formal analyses. Goldberg (2009) notes the TBM as an exciting model that can provide important insights as to how constructions are learned and produced, while Blevins and Blevins (2009) refer to it as a highly successful model for the early acquisition of syntax (see also Ambridge & Lieven, 2011). Indeed, the results of our implementation were positive and supportive of the general capabilities of the TBM model in providing a plausible account of the underlying processes of language acquisition, shedding light on more general questions raised in prior research, such as the impact of corpus density and the comparison between CDS and CS as input for the model.

However, our results also reveal several drawbacks of the TBM. First, we found a significant difficulty in integrating the versions of the TBM as these are presented in the various papers making use of the model. One major source of variation between the papers is the way slots are defined: instead of relying on lexical material, Dąbrowska and Lieven (2005) introduce abstract Component Units. Several questions arise regarding how these units are treated in the TBM studies. Apart from the units defined as THING (or REFERENT) and PROCESS, which are extensively motivated in Dąbrowska (2000), no consistent and consistently detailed definition is provided for the other types of unit. Nor is there sufficient detail of the procedure involved in identifying the various component units. Besides, the number and labels of the units themselves change from one study to the next, and it is clear that they do not constitute a comprehensive list of all possible units. The productive units in this model are thus only partially defined.

Additional significant variation between the papers of the TBM was found in the number of operations used in each study. This makes it difficult to decide which configuration is most efficient. Nor is there any comparison of the different sets of operations or an assessment of the varying levels of success in deriving novel utterances, although such differences clearly emerge from considering the percentages retrieved by

each test (in this, reliance on the same set of data is beneficial). Moreover, the difference between SUBSTITUTE and SUPERIMPOSITION is not defined clearly enough to allow for separate treatment (at one point, they are defined as two instances of the same operation). Finally, the constraints on the JUXTAPOSITION or ADD operation seem to be problematic: How is a child to recognize which elements do and do not participate in this operation? What is the underlying mechanism that constrains them as such? Besides, only vocatives and adverbials are mentioned as (prototypical) instances of such an operation, although discourse markers that bear the same form of conjunctions and that can appear at both ends of an utterance (Mulder, Thompson & Williams, 2009) can arguably be considered together with these elements.

This issue reflects on DL's own criticism of the TBM. Dąbrowska and Lieven (2005) claim that the procedure suggested by Lieven *et al.* (2003) 'is too unconstrained since the five operations defined by the authors made it possible, in principle, to derive any utterance from any string' (p. 439). Our implementation clearly shows that even the application of only two operations can still highly overgenerate. We suggest that the source of this overgeneration is not only the still imprecise definitions on which the model relies but also additional factors that have to date not been integrated into the model. Thus, not only the absence of a clear methodology for the implementation of the various operations but also the seemingly limitless number and type of operations allowed by the TBM, the lack of consideration of specific frequency effects and of the order of utterances in general, all contribute to the analysis of ungrammatical utterances.

This article also underscores the lack of an accepted methodology for the evaluation of models of language acquisition (Zaanen & Geertzen, 2008). Much work still needs to be done in this area: for example, analyzing the reverse utterances of a given corpus would be a less suitable procedure in a free-word-order language, unlike English. Using perplexity as a measure of fitness of some (language) model to test data is an attractive idea, but, as noted above, such a measure would have to be adapted for use in researching child language, where utterances are typically very short.

The analysis presented here underlines the need for more rigorous formulation of computational models of language acquisition, so as to allow for limiting their expressive power: psycholinguistic research suggests that early language is highly constrained, that utterances are short and repetitive (Brodsky, Waterfall & Edelman, 2007) and that deeply nested structures emerge later, and even then are very constrained (Bannard *et al.*, 2009). Highly expressive models such as the ones employed here are likely to overgenerate, while computational models that induce unrestricted context-free grammars from the data (Bannard *et al.*, 2009; Solan *et al.*, 2005) will fall into the same trap.

Having said that, this article is not a criticism of the TBM method per se. Rather, we wish to emphasize the importance of rigorous, robust computational evaluation of cognitive models and methods. Only with such an evaluation can design limitations and inconsistencies be found and addressed. We thus see this as an opportunity to contribute to the discussion of the type of computational model that is more suitable to the task of representing language acquisition processes. Much of the research in computational grammar induction is dedicated to learning expressive models, typically context-free grammars. We are currently investigating a much more constrained model, based on a restricted variant of finite-state automata, that we believe could account for the type of generalizations exhibited by early language learners without resorting to the over-generalization we point to in this article.

## REFERENCES

Alishahi, A. (2010). *Computational modeling of human language acquisition*. San Fransisco: Morgan & Claypool.

Alishahi, A. & Stevenson, S. (2008). A computational model of early argument structure acquisition. *Cognitive Science* **32**(5), 789–834.

Ambridge, B. & Lieven, E. V. M. (2011). *Child language acquisition: contrasting theoretical approaches*. Cambridge: Cambridge University Press.

Bannard, C. & Lieven, E. (2009). Repetition and reuse in child language learning. In R. Corrigan, E. Moravcsik, H. Ouali & K. Wheatley (eds.), *Formulaic language*, 297–321. Amsterdam: John Benjamins.

Bannard, C., Lieven, E. & Tomasello, M. (2009). Early grammatical development is piecemeal and lexically specific. In *Proceedings of the National Academy of Science* **106**(41), 17284–89.

Bates, E. & MacWhinney, B. (1987). Competition, variation, and language learning. In B. MacWhinney (ed.), *Mechanisms of language acquisition*, 157–93. Hillsdale, NJ: Lawrence Erlbaum Associates.

Berant, J., Gross, Y., Mussel, M., Sandbank, B. & Edelman, S. (2007). Boosting unsupervised grammar induction by splitting complex sentences on function words. In *Proceedings of the 31st Boston University Conference on Language Development*, 93–104. Somerville, MA: Cascadilla Press.

Blevins, J. P. & Blevins, J. (2009). Introduction: analogy in grammar. In J. P. Blevins & J. Blevins (eds.), *Analogy in grammar: form and acquisition*, 1–12. Oxford: Oxford University Press.

Bod, R. (2009a). Constructions at work or at rest? *Cognitive Linguistics* **20**(1), 129–34.

Bod, R. (2009b). From exemplar to grammar: a probabilistic analogy-based model of language learning. *Cognitive Science* **33**(5), 752–93.

Borensztajn, G., Zuidema, W. & Bod, R. (2009). Children's grammars grow more abstract with age — evidence from an automatic procedure for identifying the productive units of language. *Topics in Cognitive Science* **1**, 175–88.

Brodsky, P., Waterfall, H. & Edelman, S. (2007). Characterizing motherese: on the computational structure of child-directed language. In *Proceedings of the 29th Cognitive Science Society Conference*. Austin, TX: Cognitive Science Society.

Brown, R. (1973). *A first language: the early stages*. Cambridge, MA: Harvard University Press.

Bybee, J. (1995). Regular morphology and the lexicon. *Language and Cognitive Processes* **10**(5), 425–55.
Bybee, J. (2006). From usage to grammar: the mind's response to repetition. *Language* **82**(4), 711–33.
Chang, F. & Fitz, H. (forthcoming). Computational models of sentence production: a dual-path approach. In V. Ferreira, M. Goldrick & M. Miozzo (eds.), *The Oxford handbook of language production*. Oxford: Oxford University Press.
Chang, F., Lieven, E. & Tomasello, M. (2008). Automatic evaluation of syntactic learners in typologically-different languages. *Cognitive Systems Research* **9**(3), 198–213.
Chang, N. (2008). Constructing grammar: a computational model of the emergence of early constructions. Unpublished doctoral dissertation, Computer Science Division, University of California at Berkeley.
Chater, N. & Manning, C. D. (2006). Probabilistic models of language processing and acquisition. *Trends in Cognitive Sciences* **10**(7), 335–44.
Chater, N. & Redington, M. (1999). Connectionism, theories of learning, and syntax acquisition: where do we stand? *Journal of Child Language* **26**(1), 217–60.
Christiansen, M. H. & MacDonald, M. C. (2009). A usage-based approach to recursion in sentence processing. *Language Learning* **59**, 126–61.
Dąbrowska, E. (2000). From formula to schema: the acquisition of English questions. *Cognitive Linguistics* **11**(1/2), 83–102.
Dąbrowska, E. & Lieven, E. (2005). Towards a lexically specific grammar of children's question constructions. *Cognitive Linguistics* **16**(3), 437–74.
Demuth, K. (2008). Exploiting corpora for language acquisition research. In H. Behrens (ed.), *Corpora in language acquisition research: history, methods, perspectives*, 199–205. Amsterdam: John Benjamins.
Freudenthal, D., Pine, J. M. & Gobet, F. (2006). Modelling the development of children's use of Optional Infinitives in Dutch and English using MOSAIC. *Cognitive Science* **30**, 277–310.
Freudenthal, D., Pine, J. M. & Gobet, F. (2007). Understanding the developmental dynamics of subject omission: the role of processing limitations in learning. *Journal of Child Language* **34**(1), 83–110.
Freudenthal, D., Pine, J. M. & Gobet, F. (2009). Simulating the referential properties of Dutch, German, and English root infinitives in MOSAIC. *Language Learning and Development* **5**, 1–29.
Freudenthal, D., Pine, J. M. & Gobet, F. (2010). Explaining quantitative variation in the rate of Optional Infinitive errors across languages: a comparison of MOSAIC and the Variational Learning Model. *Journal of Child Language* **37**(3), 643–69.
Goldberg, A. (2006). *Constructions at work: the nature of generalization in language*. Oxford: Oxford University Press.
Goldberg, A. (2009). Constructions work. *Cognitive Linguistics* **20**(1), 201–224.
Lewis, J. B. & Elman, J. L. (2001). A connectionist investigation of linguistic arguments from poverty of the stimulus: learning the unlearnable. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*, 552–57. Mahwah, NJ: Lawrence Erlbaum.
Lieven, E., Behrens, H., Speares, J. & Tomasello, M. (2003). Early syntactic creativity: a usage-based approach. *Journal of Child Language* **30**(2), 333–70.
Lieven, E., Pine, J. M. & Baldwin, G. (1997). Lexically-based learning and early grammatical development. *Journal of Child Language* **24**(1), 187–219.
Lieven, E., Salomo, D. & Tomasello, M. (2009). Two-year-old children's production of multiword utterances: a usage-based analysis. *Cognitive Linguistics* **20**(3), 481–507.
MacWhinney, B. (1975). Rules, rote, and analogy in morphological formations by Hungarian children. *Journal of Child Language* **2**, 65–77.
MacWhinney, B. (ed.) (1999). *The emergence of language*. Mahwah, NJ: Lawrence Erlbaum Associates.
MacWhinney, B. (2000). *The CHILDES Project: tools for analyzing talk*, 3rd edn. Mahwah, NJ: Lawrence Erlbaum Associates.

Mulder, J., Thompson, S. A. & Williams, C. P. (2009). Final *but* in Australian English conversation. In Pam Peters, Peter Collins & Adam Smith (eds.), *Comparative studies in Australian and New Zealand English: grammar and beyond*, 337–58. Amsterdam: John Benjamins.

Parisien, C., Fazly, A. & Stevenson, S. (2008). An incremental Bayesian model for learning syntactic categories. In *Proceedings of the Twelfth Conference on Computational Natural Language Learning*, 89–96. Stroudsburg, PA: Association for Computational Linguistics.

Peters, A. M. (1983). *The units of language acquisition*. New York: Cambridge University Press.

Reali, F., Christiansen, M. H. & Monaghan, P. (2003). Phonological and distributional cues in syntax acquisition: scaling up the connectionist approach to multiple-cue integration. In *Proceedings of the 25th Annual Conference of the Cognitive Science Society*, 970–75. Boston, MA: Cognitive Science Society.

Redington, M., Crater, N. & Finch, S. (1998). Distributional information: a powerful cue for acquiring syntactic categories. *Cognitive Science* **22**(4), 425–69.

Rowland, C. F., Fletcher, S. L. & Freudenthal, D. (2008). How big is big enough? Assessing the reliability of data from naturalistic samples. In H. Behrens (ed.), *Corpora in language acquisition research: history, methods, perspectives*, Vol. 6, 1–24. Amsterdam: John Benjamins.

Sagae, K., Davis, E., Lavie, A., MacWhinney, B. & Wintner, S. (2010). Morphosyntactic annotation of CHILDES transcripts. *Journal of Child Language* **37**(3), 705–729.

Solan, Z., Horn, D., Ruppin, E. & Edelman, S. (2005). Unsupervised learning of natural languages. In *Proceedings of the National Academy of Sciences of the United States of America* **102**(33), 11629–34.

Suppes, P. (1974). The semantics of children's language. *American Psychologist* **29**, 103–114.

Tomasello, M. (2003). *Constructing a language*. Cambridge, MA, London: Harvard University Press.

Tomasello, M. (2006). Acquiring linguistic constructions. In D. Kuhn & R. Siegler (eds.), *Handbook of child psychology*, 255–98. New York: Wiley.

Vogt, P. & Lieven, E. (2010). Verifying theories of language acquisition using computer models of language evolution. *Adaptive Behavior* **18**(1), 21–35.

Zaanen, M. van & Geertzen, J. (2008). Problems with evaluation of unsupervised empirical grammatical inference systems. In *ICGI '08: Proceedings of the 9th International Colloquium on Grammatical Inference*, 301–303. Berlin, Heidelberg: Springer-Verlag.