Extending the Past-tense Debate: a Model of the German Plural

Niels A. Taatgen (niels@ai.rug.nl)
Artificial Intelligence, University of Groningen
Grote Kruisstraat 2/1, 9712 TS Groningen, the Netherlands

Abstract

One of the phenomena that has been studied extensively in cognitive science is learning the English past tense. Many models have been made of the characteristic U-shape in performance on irregular verbs during development. An important test case for such models is whether they can be extended to other examples of inflection. A case that is often quoted as particularly tough is the German plural. In the present study, an ACT-R model of the past tense is applied to the German plural. The model not only successfully learns the default rule, but also exhibits some other characteristics of the German plural.

Introduction

Learning the English past tense has been one of the central topics of debate in cognitive science since McClelland and Rumelhart published their original neural network model in 1986. The phenomenon is very simple. English verbs can be broken down into two categories: regular and irregular verbs. The past tense of a regular verb can be obtained by simply adding *-ed* to the stem. Irregular verbs on the other hand are unsystematic: each verb has a unique inflection. When children have to learn the inflection of the past tense, they go through three stages. In the first stage their use of the past tense is infrequent, but when they use the past tense they do so correctly. In the second stage they use the past tense more often, but they start overregularizing the irregular verbs. So instead of saying *broke*, they now say **breaked*. On the other hand, inflection of regular verbs increases dramatically, indicating that the child has somehow learned the general regular pattern. In the third stage, they inflect irregular verbs correctly again. This pattern of learning is often referred to as U-shaped learning.

Although learning the past tense seems to be a rather simple problem, it nevertheless encompasses a number of issues in language acquisition and learning in general. Apparently the past tense has two aspects: on the one hand there is a general rule, and on the other hand there is set of exceptions. Children are able to learn both aspects, and the phenomenon of U-shaped learning seems to implicate that children learn the general rule in stage 2. The important point McClelland and Rumelhart make is that this does not necessarily imply that this knowledge is actually represented as a rule in the cognitive system: their neural network model has no separate store for rules, but it nevertheless exhibits rule-like behavior in the form of U-shaped learning. Ever since their original model, the neural network approach has been challenged (e.g., Pinker & Prince, 1988), improved (e.g., Plunkett & Marchman, 1991), challenged again (e.g., Marcus, 1995) and improved again (e.g., Plunkett & Juola, 1999). I would like to highlight two unresolved issues in this debate, because they will be addressed here. The first issue is feedback. A well-known fact in language acquisition is that children do not rely on feedback on their own production of language (at least with respect to syntax), simply because they do not receive any (Pinker, 1984). Although this problem is addressed by some modelers (e.g., Plunkett & Juola, 1999), its resolution is not entirely satisfactory: the assumption is that learning takes place while children perceive past tenses, and not while they actually produce past tenses. This idea is at odds with the picture of skill acquisition in general, where practice is considered as a main means of learning. A second issue is the frequency of the regular cases. In English, most verbs are regular. This fact is essential for neural network models, as they need to be presented with regular cases at least 50% of the time (Marcus, 1995). This is already slightly problematic in English, as the token-frequency of regular verbs, how often a verb is actually used in language, is only around 30% (irregular verbs are just used much more often than regulars). Connectionist modelers have therefore introduced the input/uptake distinction: not every word that is perceived is presented to the network. This assumption becomes especially problematic if regular forms are much more rare. An example of inflection where the regular form is very rare is the German plural.

The German Plural

German has five different suffixes to mark plurality of a noun: zero (no suffix), *-(e)n*, *-e*, *-er* and *-s*. Moreover, the stem-vowel sometimes receives an Umlaut (¨), something we will ignore for the present. The plural is almost always indicated by suffixation: there are only a few exceptions, mainly words derived from Latin (e.g., *Thema-Themen*). Careful analysis of these suffixes has revealed that the *-s* suffix is actually the default rule (Marcus, Brinkmann, Clahsen, Wiese & Pinker, 1995). Interestingly enough, this suffix is also the least frequent of all five, both in *type-frequency* (how many words are there) and *token-frequency* (how often are they used). Marcus et al. estimate the type frequency of nouns ending in *-s* at 4%, and the token frequency at only 2%. It appears however that at least some of the other suf-

fixes are somehow tied up by additional constraints: for example, zero and *-er* are never used for feminine words, *-e* cannot be used if the stem already ends with and *e*, etc.

The combination of a default rule that is based on very low frequencies and the fact that there is no feedback on production makes it very hard to understand how the default rule can be learned at all. If there is no feedback, the cognitive system has to construct its own language based on perceived inputs from the environment. But why would the cognitive system always elect to use *-s* as a default rule, while there are other options as well?

A possible source of information on this topic is to look at children, and examine the type of errors they make. Marcus et al. (1995) quote a number of studies that indicate children overregularize using the *-s* suffix (in 10-15% of the opportunities), although it is not the most common overregularization (*-(e)n* is the most common overregularization). This pattern is similar to the English past tense, which has been studied much more extensively, except that in English the default rule is also the dominant source of overregularization.

Regular versus irregular inflection is often characterized by competition between a rule and exceptions. It appears that in the case of the German plural there is also competition among rules, a competition the *-s* rule eventually wins.

To summarize, the German plural is in some sense similar to the English past tense, but more complicated. There is competition among candidate rules, while the English past tense has only one apparent candidate rule, and the eventual rule is based on nouns that have a low frequency, as opposed to the high frequency of regular English verbs. The German plural is therefore an interesting test case for existing models of the past tense in English: can these models successfully account for the German plural as well? Taatgen and Anderson (submitted) developed a model for the English past tense, the present study will show it is extendable to the German plural as well, without many modifications.

Towards a General Model of Regular and Irregular Inflection

The Taatgen and Anderson (submitted) model of learning the past tense is based on the ACT-R architecture (Anderson & Lebiere, 1998). It is a so-called dual-representation model, so it separately represents examples and rules, corresponding to ACT-R's declarative chunks and procedural rules. Although two representations make the model weaker than neural network models that use only one type of representation, it does not need a number of assumptions neural network models need. The ACT-R model does not need a specific input regimen, in which the vocabulary is gradually increased. Neural network models typically start training with a small set of words, in the order of 10 to 20. This number is increased during training. A problem with this approach is that the way in which the input-set is increased can be manipulated to get the desired outcome. The ACT-R model on the other hand can be trained on the full vocabulary right from the start, and with the exact token frequencies as in normal language.

A second advantage of the ACT-R model is that it can learn without feedback on its own performance. The model assumes examples of past tenses are perceived and stored in declarative memory, and that no feedback is given on production. The only feedback the model uses when it produces language is its internal feedback: the effort production took. It will prefer strategies that take the least effort, as opposed to strategies that produce the right answer (as it has no way of knowing what the right answer is). Neural networks have to make additional assumptions to account for the lack of feedback. As a neural net needs to know the correct answer to adjust its weights, it has to learn during the perception of language instead of during production. It is assumed that once an inflected form is analyzed, it is "recreated" by a hypothesis generator. To quote Plunkett and Juola (1999):

> The child is continually taking in word tokens and comparing the words actually heard (e.g., "went") to the tokens that the child's hypothesis generator would have expected to produce as inflected forms of a given stem; when they differ, this provides evidence to the child that the hypotheses are wrong and should be modified. (p. 466)

Although this view on language acquisition is not necessarily false, there is no clear evidence for it, and must be considered as an extra assumption. To summarize, although neural network models need only one form of representation, and are stronger theories in that respect, they are weaker with respect to the shape of the input and the organization of feedback.

One of the main claims of cognitive modeling is that models can be generalized to other tasks and contexts. A model of the past tense should be a stepping stone towards a more general model of regular and irregular inflection in different languages. The first step is discussed in this paper: the German plural. Before I will discuss the model, I will briefly explain a few relevant ACT-R aspects.

Rules and Examples in ACT-R

According to the ACT-R theory and architecture (Anderson & Lebiere, 1998) human memory consists of two long-term stores: a declarative memory and a procedural memory. As ACT-R is a hybrid architecture, representations in both memory systems have symbolic and subsymbolic aspects.

Declarative memory is used to store facts, goals and perceptual information. For the purposes of the model, it will store the words in the vocabulary, and examples of how an inflected form, in this case the plural form, is constructed. The declarative memory may store the fact (called a *chunk*) that "Jahr" (year) is a noun, and that "Jahre" is the plural of "Jahr". Declarative memory may contain false facts along-

side true facts, so it may have a chunk "Jahr"-"Jahren" as well as the correct chunk. Each chunk has an activation value, that represents the log odds that the chunk will be needed in the current context. So ACT-R doesn't really care about truth although true facts are probably more often needed than false facts.

For the purpose of the current model, the main determiner of the activation value is repetition. If a certain chunk is retrieved often from memory, or is perceived often, its activation value will be high. The main effect of activation for the present model is that chunks whose activation is too low cannot be retrieved from memory. Also, if two or more chunks are candidates for retrieval at the same time, the chunk with the highest activation is chosen (more precisely: has the highest probability of being chosen as noise is added to the process).

Chunks cannot act by themselves, they need production rules from procedural memory for their application. In order to use a chunk, a production rule has to be invoked that retrieves it from declarative memory and does something with it. Since ACT-R is a goal-driven theory, chunks are always retrieved to achieve some sort of goal. In the context of inflection of words the goal is simple: given the stem of a word, produce the proper inflection. One strategy to produce a certain inflection is to just retrieve it from declarative memory, using a production rule like:

```
IF    the goal is to produce a certain
      inflection of a word
      AND there is a chunk that specifies this
      inflection for that word
THEN  the set the answer of the goal
      to that inflected form
```

If the goal is to produce the plural of a certain word, this production rule will attempt to retrieve a chunk from declarative memory that specifies what the plural is of that word. Of course this production rule will only be successful if such a chunk is present and its activation is high enough.

The behavior of production rules is also governed by subsymbolic parameters. Each production rule has some real-parameters associated with it that estimate its expected outcome. This expected outcome is calculated from estimates of the cost (in time) and probability of reaching the goal if that production rule is chosen. The unit of cost in ACT-R is time. ACT-R's learning mechanisms constantly update these estimates based on experience. If multiple production rules are applicable for a certain goal, the production rule is selected with the highest expected outcome. This is again a noisy process, so the rule with the highest expected gain only has the highest probability of being selected first. If a rule is selected and subsequently fails, the next best rule is tried. For example, if the example rule above fails to retrieve a chunk that specifies the inflected form, the next best rule will be tried.

New rules are learned through a process of specialization and compilation. In the present model, this process will specialize the general strategy of analogy into specific rules for inflection[1]. I will discuss the details of this process later in the paper.

The Model of the German Plural

Prior knowledge in the model

The model starts out with a set of general problem-solving strategies that are often used in ACT-R models:

1. **Retrieval**. A general strategy for problem solving is to search declarative memory for a case that is identical to the case at hand. If an identical case can be found, it immediately produces the answer, else one of the other strategies is tried.
2. **Analogy**. Another general strategy is to look for a case that is similar to the current problem. After a suitable example has been found, a mapping has to be found between the problem and the answer in the example. This mapping is then applied to the case at hand. In the present model, an arbitrary case of a plural noun is retrieved from declarative memory, after which very simple pattern-matching productions identify the mapping and apply it to the current noun.
3. **Do nothing**. A third strategy is to just do nothing. This strategy is not as stupid as it sounds, as some problems may not be worthwhile solving as long as no additional knowledge is available. The result of doing nothing is that the singular form will be used instead of the plural.

Input

The CELEX database was used to select the 538 most frequent German nouns. After removing the nouns that have no plural, 472 nouns were left. The frequencies of the plural forms are in Table 1. Note that the frequencies for the *-s* suffix are even lower than the Marcus et al. estimates, probably because only high frequency words were selected.

Table 1: Frequencies of suffixation in the CELEX sample

Suffix	Type frequency	Token frequency
-(e)n	48%	50%
-e	34%	35%
zero	11%	8%
-er	5%	7%
-s	1.3%	1%
other	1%	0.4%

To simulate a child's perception and production of plurals, the following input regimen for the model is used: every 2000 seconds, one word is randomly drawn from the set of

1. The process of proceduralization used in this model is not part of ACT-R 4.0, the current version of ACT-R, but is part of a proposal for the next version of the architecture

472 words, based on the token frequency of the word. The model has to produce the plural of this word, simulation production by child. As token frequencies are used, high-frequency words are drawn more often, in the same proportion as they occur in the corpus. Also, every 2000 seconds two random plurals, drawn in the same manner, are added to declarative memory, reflecting perception from the outside world.

Learning

During the simulation, several learning processes influence the behavior of the model, from symbolic to subsymbolic and from declarative to procedural. Table 2 summarizes the different learning mechanisms.

Table 2: Learning mechanisms and their effects

Type of learning	**Effect on the model**
Declarative Symbolic	Examples of plurals are added to memory. Examples are perceived in the environment, and produced by the model itself.
Declarative Subsymbolic	Examples that occur often or are retrieved often are more readily available in memory, as they receive a higher activation. Low-activation examples cannot be retrieved.
Procedural Symbolic	Rules are learned to add specific suffixes to the stem in order to create a plural.
Procedural Subsymbolic	The expected gain of each strategy is estimated based on experience: the rule that takes the least effort to produce a plural is favored.

A first aspect of learning is that new production rules are learned that add the different suffixes to the stems. These rules are learned by specializing analogy. Analogy can be characterized by two steps: retrieving an example from declarative memory and applying this example to the new case. Proceduralization eliminates the retrieval of the example, and substitutes variables in the rules with a certain example. It then combines the two steps in a single step. The result is a rule that approximately acts like analogy, but always with the same example. In the case of inflection, the suffix of the retrieved example determines what the new rule will do: if the example has an *-e* suffix, the new rule will always add the *-e* suffix to produce a plural.

A second procedural aspect of learning is that rules compete. The three strategies mentioned before, together with the rules that proceduralization produces, all compete in producing an inflection. Although they do not receive feedback whether what they produce is correct, they do receive feedback on how much effort it took to produce an inflection. This effort can be influenced by many factors. For example, if the retrieval rule fails to find an example, another strategy has to be tried afterwards, increasing the average effort of retrieval. If a rule produces a suffix that is long to pronounce, using that strategy takes more effort than a short suffix. With respect to this pronunciation effort, the model assumes that using a suffix implies some extra effort. It shares this assumption with the past tense model. Moreover, it assumes the *-s* suffix takes slightly less effort than the other suffixes, because *-s* suffix is just an additional phoneme, while the other suffixes are extra syllables. This is an important assumption, because it will be the main reason why the *s*-suffixation rule will eventually dominate other suffixation rules.

Learning in declarative memory also plays a key role in this model. On the symbolic level, examples of past tenses are constantly added to memory, by perceiving them in the outside world, but also by producing them. The fact that an example is in declarative memory does not guarantee that it can be retrieved. This is where the subsymbolic level is important: activation decays with time, making the example irretrievable. Another aspect of activation is to decide in the case of multiple choices: if two chunks match, the chunk with the highest activation is chosen.

It is important to note than whether or not a produced plural is correct has no impact on the learning. Correctness only plays a role in the examples that the model perceives in the world, but even there an occasional error will not disrupt performance.

Results of the Model

The model was run for 80000 trials, or slightly over 60 simulated months. Figure 1 shows the expected gains of the different rules. Remember that the rule with the highest expected gain is generally tried first, and if it fails the next best rule is tried (although noise may change the order from time to time). Right from the start, retrieval is the dominant strategy. Its expected gain improves quickly as more and more examples are learned. Retrieval is not always successful, so the order of the remainder of the rules is especially important. The rules for the zero, *-e*, *-(e)n* and *-er* suffixes are learned very early in the simulation, and appear to be reasonably productive, as they pass both the do-nothing and the analogy strategy around month 5. Only after 10 months in the simulation the *-s* rule is learned, due to the fact that its occurrences and therefore the opportunities for generalization are rare. Once the *-s* rule is learned, however, it quickly dominates the earlier suffixation rules due to its pronunciation advantage.

The expected gains of the rules have a direct impact on the performance of the model, depicted in Figure 2. As the expected gain of the retrieval-rule increases, the proportion of correct responses also increases (Figure 2a). When after

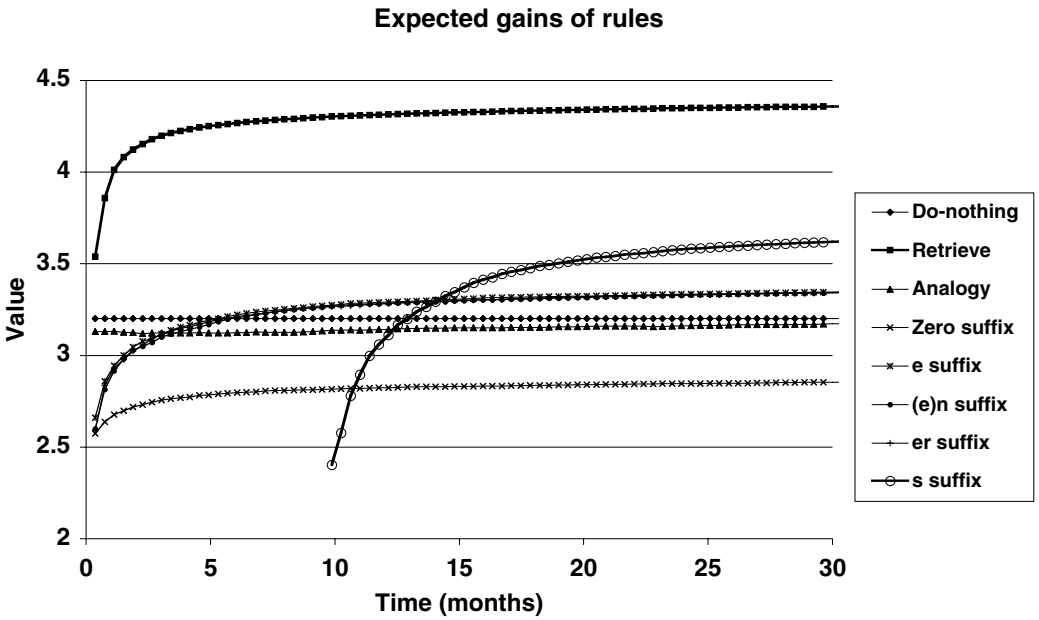


Figure 1: Expected Gains of the different strategies during the first 30 months of the simulation

10 months the *-s* rule is learned, the model starts making errors by adding it to other stems.

Figure 2b shows all the errors that the model makes: at the start of the simulation, errors are dominated by the "do-nothing" rule (producing just stem, so a zero-suffix), as very few plurals are yet known. Soon afterwards, the rules for *-(e)n*, *-e* and *-er* are learned, so they dominate the errors. After month 10, the rule for *-s* dominates the errors, although other errors are still made due to noise, and retrieval of past errors. If we extrapolate the results presented here towards adulthood, the dominant strategy will be to retrieve the plural form from memory. If that process fails, the rules that add the *-s* suffix will be used: exactly what one would expect from a default rule.

Discussion

The present model shows that the original Taatgen and Anderson (submitted) model of the English past tense can be extended to the German plural without modifications. The model is able to learn the default rule despite the low incidence of examples, which is an important problem for other models. But how well does the model fit the data? Unfortunately, the data on the German plural is not as extensive as data on the English past tense.

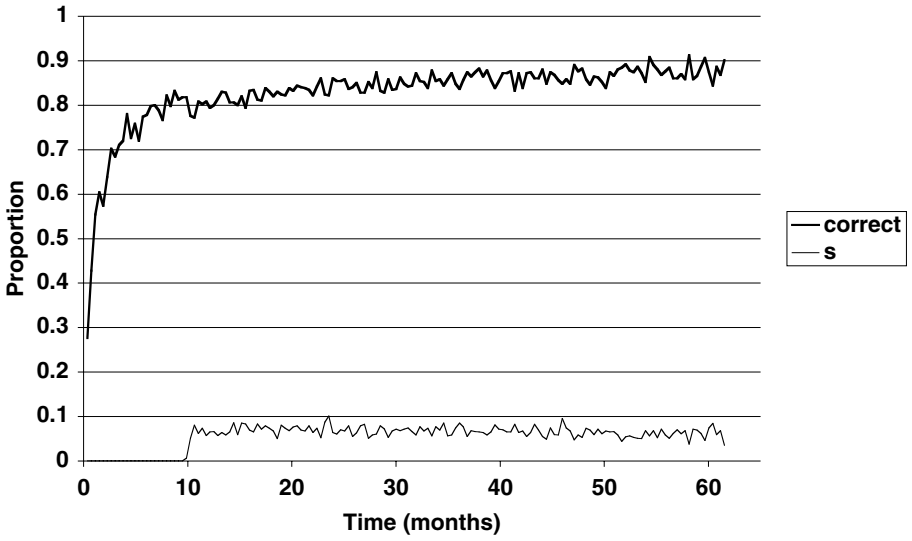
The basic facts from Marcus et al. (1995) are that German children overregularize by using the *-s* suffix, but also by using other suffixes like *-(e)n*. Also, German children make much more errors in the plural (Marcus et al. quote percentages between 11% and 25% for just the *-s* overregularization) than English children with the past tense (usually less than 10%). Both these facts are supported by the model. The fact that retrieval is the dominant strategy implies that rules play only a minor role in inflection, and that this role is most important during the learning phase, when not all inflected forms are memorized yet. In English, the regular rule for the past tense leads to reasonable performance, as the majority of the verbs is regular. In German, however, the regular rules will generally not produce correct behavior, so it is no surprise children make many errors.

The present model operates on a rather global level, largely ignoring some issues concerning phonetics and gender. Obviously, the *-e* suffix cannot be used when the stem already ends with an *-e*. This would make the *-e* rule less attractive, as it will sometimes fail. Gender may also place additional constraints on certain rules. Furthermore, due to phonological constraints, it is sometimes necessary to add the Umlaut to the vowel in the stem with certain suffixes. The *-s* suffix is free of all these constraints, and can in principle be applied in all cases. Together with the fact that it is also the shortest suffix, this makes the rule the most attractive one, despite its low incidence.

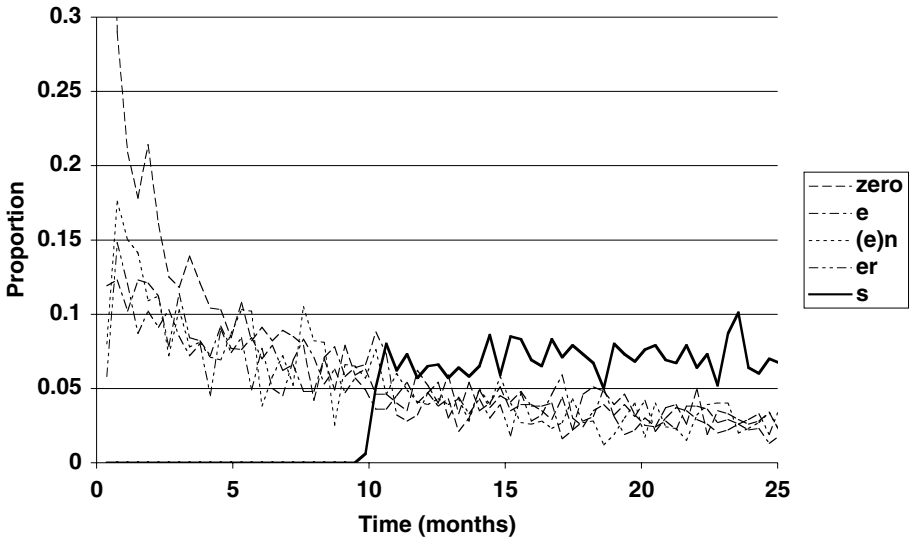
Despite the fact that it largely ignores some of the low-level details, this model demonstrates how generalization in language acquisition can be explained in the absence of feedback. Although it does not solve the learnability problem in language, it nevertheless points at a different source of feedback that may play a role in different areas of language acquisition as well: internal feedback that is not based on the correctness of the produced utterance, but based on the amount of effort it took to produce the utterance.

Acknowledgments

I would like to thank Jack Hoeksema for help in consulting the CELEX database.



(a)



(b)

Figure 2: Performance of the model (a) proportion of correct responses, and overregularization errors with the *-s* suffix (b) proportions of all errors in the first 25 months of the simulation.

References

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Erlbaum.

Marcus, G. F. (1995). The acquisition of the English past tense in children and multilayered connectionist networks. *Cognition*, *56*, 271-279.

Marcus, G. F., Brinkmann, U., Clahsen, H., Wiese, R., & Pinker, S. (1995). German inflection: the exception that proves the rule. *Cognitive Psychology*, *29*, 189-256.

Pinker, S. (1984). *Language learnability and language development*. Cambridge, MA: Harvard University Press.

Pinker, S., & Prince, A. (1988). On language and connectionism: Analysis of a distributed processing model of language acquisition. *Cognition*, *28*, 73-193.

Plunkett, K. & Juola, P. (1999). A connectionist model of English past tense and plural morphology. *Cognitive Science*, *23*, 463-490.

Plunkett, K., & Marchman, V. (1991). U-shaped learning and frequency effects in a multi-layered perceptron: Implications for child language acquisition. *Cognition*, *38*, 43-102.

Rumelhart, D. E., & McClelland, J. L. (1986). On learning the past tense of English verbs. In J.L. McClelland & D.E. Rumelhart (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition* (pp. 216-271). Cambridge, MA: MIT Press.

Taatgen, N.A. & Anderson, J.R. (submitted). Why do children learn to say "Broke"? A model of learning the past tense without feedback. Prepublication available on-line: http://ai.rug.nl/prepublications/prepubsTCW-2000-9.pdf