Available online at www.sciencedirect.com
SCIENCE DIRECT®
Brain and Language 90 (2004) 287–298
Brain and Language
www.elsevier.com/locate/b&l

Do type and token effects reflect different mechanisms? Connectionist modeling of Dutch past-tense formation and final devoicing

Fermín Moscoso del Prado Martín,* Mirjam Ernestus, and R. Harald Baayen

Max Planck Institute for Psycholinguistics, University of Nijmegen, 6500 AH Nijmegen, The Netherlands

Accepted 3 December 2003
Available online 27 February 2004

Abstract

In this paper, we show that both token and type-based effects in lexical processing can result from a single, token-based, system, and therefore, do not necessarily reflect different levels of processing. We report three Simple Recurrent Networks modeling Dutch past-tense formation. These networks show token-based frequency effects and type-based analogical effects closely matching the behavior of human participants when producing past-tense forms for both existing verbs and pseudo-verbs. The third network covers the full vocabulary of Dutch, without imposing predefined linguistic structure on the input or output words.

Keywords: Connectionist model; Past-tense formation; Simple recurrent networks; Type-based analogical effects; Token-based effects; Orthography; Final devoicing; Dutch; Accumulation of expectations

1. Introduction

Dutch regular past-tenses are formed by adding *de* (/də/) or *te* (/tə/) to the verbal stem. The choice of the allomorph depends on the last phoneme of the stem by a simple rule: Verb stems ending in an underlyingly unvoiced obstruent take *te*, while all other verbs take *de*. For instance, as the stem of the verb *harken*, /hɑrkən/, "to rake" is /hɑrk/ ending in a unvoiced /k/, its singular past-tense form is *harkte*, /hɑrktə/, "raked". Similarly, the verb *zorgen*, /zɔrɣən/, "to care", with its stem /zɔrɣ/, underlyingly ending in voiced /ɣ/, has the singular past-tense form *zorgde*, /zɔrɣdə/, "cared".

Final devoicing introduces a complication to the rule of Dutch past-tense formation. In Dutch, all obstruents in word-final positions are realized as unvoiced (except before voiced plosives), independently of their underlying voice specification. This makes it impossible to infer the underlying voice specification of word-final obstruents from their acoustic realization. For example, due to final devoicing, the acoustic form [ɪk laːt] could be the first person singular form of either the verb *laden*, /laːdən/, "to load," or *laten*, /laːtən/, "to let". As a consequence, it is impossible to know which is the correct regular past-tense allomorph for a new or unknown Dutch obstruent-final verb, when the verb stem is presented auditorily without being followed by a vowel-initial suffix. Thus, the past-tense form of the pseudo-verb [dɑp] could either be *dabde* or *dapte*.

Ernestus and Baayen (2001, 2003) investigated how speakers of Dutch decide which is the past-tense form for existing verbs and pseudo-verbs. They asked Dutch participants to write down the regular past-tense forms of existing Dutch verbs and pseudo-verbs. For example, after hearing the pseudo-verb [ɪk dɑp], participants had to write down *ik dapte* or *ik dabde*, depending on which underlying voice specification they attributed to the final [p]. Similarly after hearing [ɪk dʏp] ("I doubt"), participants had to write down its correct regular past-tense form *ik dubde* ("I doubted"). Ernestus and Baayen (2003) found that Dutch speakers based their interpretation of the final obstruent of pseudo-verbs on the phonologically similar existing words. In general, for pseudo-words, participants interpret a final obstruent as voiced when the majority of the words in their lexicon

* Corresponding author. Present Address. MRC Cognition, Brain Sciences Unit, Cambridge CB2 2EF, UK. Fax: +01223-359-062.
E-mail address: fermin.moscoso-del-prado@mpi.nl (F. Moscoso del Prado Martín).

0093-934X/$ - see front matter © 2003 Published by Elsevier Inc.
doi:10.1016/j.bandl.2003.12.002

with similar phonological properties end in a voiced obstruent, and they consider it to be unvoiced in the opposite case. In the [ɪk dɑp] example, most participants opted for *te*, since most final bilabial plosives following short vowels in existing words are unvoiced. The proportion of the *de* responses to pseudo-words reflected very closely the proportion of similar words ending in voiced obstruents in the lexicon. Ernestus and Baayen describe this analogical effect as type-based, i.e., the proportion of *de* responses depends on the number of similar words ending in a voiced obstruent, independently of their frequencies of occurrence. Existing Dutch verbs were also sensitive to this analogical effect. For instance, 43% of the participants produced *ik dubte* as the past-tense form for [ɪk dʏp], by analogy with the majority of existing forms in the lexicon such as *hup*, [hʏp], *stap*, [stɑp], *klap*, [klɑp], *stop*, [stɔp], etc. In addition to this type-based analogical effect, there was a token-based effect. High frequency past-tense forms were less sensitive to the analogical effect, producing fewer errors (Ernestus & Baayen, 2001).

The presence of both a type-based analogical effect and a token-based frequency effect can be interpreted as a reflection of two separate processing mechanisms. Some authors (e.g., Clahsen, 1999; Pinker, 1999; Pinker & Prince, 1994, 1998) propose a dual route system in which one route is a symbolic rule application mechanism that deals with the regular forms, and the other is an associative memory that stores the irregular forms. In such a model only the associative memory mechanism would be sensitive to token frequency effects. The mechanism in charge of the morphological generalizations (either rules or statistical analogies) has been argued to learn on the sole basis of type frequencies (e.g., Albright & Hayes, 2003; Bybee, 1995, 2001). These theories predict the existence of a rule application or analogical mechanism operating on types, which would be in charge of producing the regular forms, in combination with a token frequency sensitive mechanism which would be used to store and retrieve the memorized forms.

In contrast to the predictions of the traditional dual route mechanism, token frequency effects have been shown to affect the processing of morphologically regular forms as well (e.g., Baayen, Dijkstra, & Schreuder, 1997; Baayen, Schreuder, De Jong, & Krott, 2002; Schreuder, De Jong, Krott, & Baayen, 1999). This finding questions the clean separation between type-based application of regularities, and token-based memory storage of irregular forms. Additionally, Moscoso del Prado Martín, Kostić, and Baayen (in press) report that an information-theoretical approach subsumes both type-based and token-based effects under a single measure of uncertainty. Although, this measure is calculated on the basis of the token frequencies, it shows properties similar to those arising from type-based counts. Taken together, these two findings lead us to question the necessity of separating token sensitive and type sensitive mechanisms, indicating that both types of effects can be consequences of uncertainty in a purely token-based approach.

Can token-based and type-based effects also arise as consequences of one single processing mechanism, in line with the so-called 'single route' theories of lexical processing (e.g., McClelland & Patterson, 2002a; Plunkett & Juola, 1999; Rumelhart & McClelland, 1986)? In the present study, we address this question by modeling the experiments described by Ernestus and Baayen (2001, 2003) using a single route model. Ernestus and Baayen (2003) already studied several, non-connectionist, single route models that account for the analogical type-based effect. None of these models accounts for the token frequency effects reported by Ernestus and Baayen (2001) for the existing verbs.

Although, previous connectionist systems have proved capable of learning morphological generalizations on the basis of token frequencies (e.g., Rumelhart & McClelland, 1986), these models have done so with restricted, very limited vocabularies, such as only monomorphemic or even monosyllabic stems. Additionally, most of the models have used training regimes in which words are not presented with their actual frequencies of occurrence, but on transformed frequency counts (e.g., the logarithm) that have the effect of mitigating the frequency differences between high and low frequency words. Moreover, most of these models use input–output templates, which impose predefined structure on their inputs, prescribing all the possible words in a language to conform to a given pattern (e.g., CCCVVCCC, etc.). These templates provide language-specific knowledge to the system, knowledge that ideally should be acquired from the input. Additionally, the templates require reclassifying and aligning the segments of the input words as onset consonants, vowels, or coda consonants, before presenting them to the system. All this built-in linguistic knowledge considerably oversimplifies the past-tense formation problem (Pinker & Ullman, 2002a).

In the present study, we describe connectionist models that deal with the full verbal past-tense formation system of Dutch. We describe three connectionist models of Dutch past-tense formation and evaluate their performance by having them produce past-tense forms for the stimuli in the experiments by Ernestus and Baayen (2001, 2003). Finally, we discuss the implications of the results of our models for theories of lexical processing.

2. Simulation 1

In the first simulation, we modeled Dutch past-tense formation with a Simple Recurrent Network (SRN; Elman, 1990, 1993). This network allows us to represent

the input as a sequence of phonemes, without word length restrictions and without any built-in assumptions about syllable structure.

Fig. 1 shows the basic architecture of the SRN that we used in our simulations. This network consists of an input layer of 15 units, each of which represents a binary phonetic feature, plus an additional 'end-of-word' bit for triggering the output, and an output layer of 26 units, representing the letters of the alphabet. Between the input and output layers, there is a small hidden layer of 10 units. The outputs of all the units in the input layer are connected to the inputs of all units in the hidden layer, and the outputs of all units in the hidden layer are connected to the inputs of all units in the output layer. There is an additional context layer, which represents a copy of the hidden layer in the previous state in time. The outputs of the units in the context layer have all-to-all connections with the inputs of the units in the hidden layer. This single recurrent loop allows the network to maintain a memory of the activation of the hidden layer in the previous time steps (Elman, 1990, 1993).

Presenting words phoneme by phoneme at the input simulates auditory input to the network. A word is represented by a sequence of phonemes presented at consecutive time steps. Each phoneme is presented by activating its corresponding phonetic features. Table 1 shows the phonetic features that we employed in our simulations. They distinguish between all Dutch phonemes, and take the phonology of Dutch into account. The feature matrix is similar to the one presented by Booij (1995) except that we replaced [coronal] by [alveolar] and [palatal] in order to be able to distinguish /s/ and /z/ from /ʃ/ and /ʒ/. We added the feature [tense] to express the difference in quality between long (tense) vowels and short (lax) vowels. Finally, we omitted the [aspiration] feature since /h/, the only phoneme for which it is positive, can be uniquely defined without it.

The network's output is also represented by a sequence, which in this case, is a sequence of letters. Although, Dutch orthography is to a large extent transparent, with a quite regular grapheme to phoneme mapping, there are some irregularities making the orthographic transcription of one sound dependent on the following sounds. As a consequence, it is impossible to synchronize the network's input with the network's output, that is, having the system output a letter right after the corresponding phoneme is presented at the input. We therefore chose to start producing the output only after the full input had been received, and we added an additional trigger bit signaling the end of the word to the input layer of phonetic features. Only after this trigger node has been activated, we began recording (and training) the network's output.

A training regime aiming to reproduce the full letter sequence at the output would impose a tremendous amount of memory load on the network. Because of this memory load, SRN's do not perform well in reproducing full Dutch words one letter at a time, starting after having received the full input form (Stoianov, 2001). As our first investigation is concerned only with the interpretation of stem-final, neutral obstruents and the choice of the regular past-tense allomorph, we reduce the memory load by training our network to produce only the last letter of the verb stem and the two letters of its past-tense allomorph. In this way, the network needs to store only those characteristics of the words that are relevant for the interpretation of the final obstruent and the choice of the past-tense allomorph, which leads to a drastic reduction in task complexity.

A model of past-tense formation can only be realistic if its training input is similar to the input that human speakers receive. It therefore has to be exposed to past-tense forms such that each form is presented a number of times that is proportional to its frequency of occurrence. Hence, in Simulation 1a, we used a token-based training strategy. In contrast, in Simulation 1b, we used a type-based training regime; that is, all verbs were presented to the network an equal number of times,

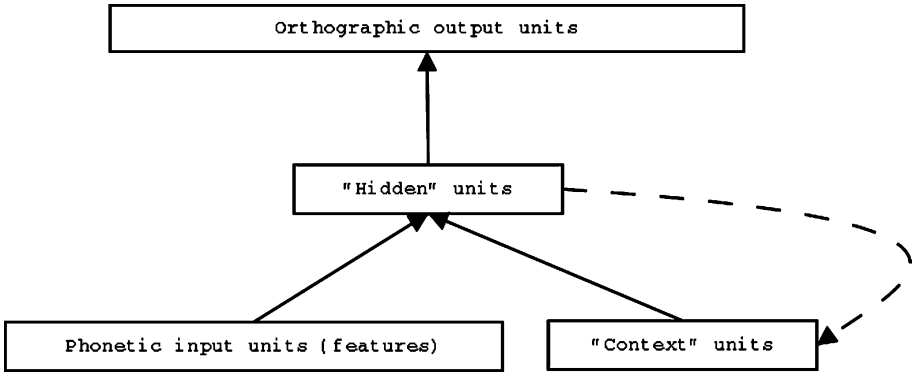

Fig. 1. Modular architecture of a Simple Recurrent Networks used in the simulations. The boxes correspond to layers of units. The solid arrows represent sets of trainable 'all-to-all' connections between the units in two layers. The dashed arrow stands for a fixed 'one-to-one' not trainable connection between two layers. These connections have the function of copying the activation of the hidden units into the context units at every time step.

Table 1
Phonetic features used for coding the input in the simulations

Phoneme	Consonant	Sonorant	Continuant	Voicing	Nasal	Approximant	Labial	Coronal	Palatal	Dorsal	Lateral	High	Mid	Back	Round	Tense
p	+	−	−	−	−	−	+	−	−	−	−	−	−	−	−	−
b	+	−	−	+	−	−	+	−	−	−	−	−	−	−	−	−
t	+	−	−	−	−	−	−	+	−	−	−	−	−	−	−	−
d	+	−	−	+	−	−	−	+	−	−	−	−	−	−	−	−
k	+	−	−	−	−	−	−	−	−	+	−	−	−	−	−	−
g	+	−	−	+	−	−	−	−	−	+	−	−	−	−	−	−
ŋ	+	+	+	+	+	−	−	−	−	+	−	−	−	−	−	−
m	+	+	+	+	+	−	+	−	−	−	−	−	−	−	−	−
n	+	+	+	+	+	−	−	+	−	−	−	−	−	−	−	−
l	+	+	+	+	−	+	−	+	−	−	+	−	−	−	−	−
r	+	+	+	+	−	+	−	+	−	−	−	−	−	−	−	−
f	+	−	+	−	−	−	+	−	−	−	−	−	−	−	−	−
v	+	−	+	+	−	−	+	−	−	−	−	−	−	−	−	−
s	+	−	+	−	−	−	−	+	−	−	−	−	−	−	−	−
z	+	−	+	+	−	−	−	+	−	−	−	−	−	−	−	−
ʃ	+	−	+	−	−	−	−	+	+	−	−	−	−	−	−	−
ʒ	+	−	+	+	−	−	−	+	+	−	−	−	−	−	−	−
x	+	−	+	−	−	−	−	−	−	+	−	−	−	−	−	−
ɣ	+	−	+	+	−	−	−	−	−	+	−	−	−	−	−	−
j	+	+	+	+	−	+	−	−	+	−	−	−	−	−	−	−
h	+	+	+	−	−	−	−	−	−	−	−	−	−	−	−	−
w	+	+	+	+	−	+	+	−	−	−	−	−	−	−	−	−
i	−	+	+	+	−	−	−	−	−	−	−	+	−	−	−	+
y	−	+	+	+	−	−	−	−	−	−	−	+	−	−	+	+
e	−	+	+	+	−	−	−	−	−	−	−	+	+	−	−	+
ø	−	+	+	+	−	−	−	−	−	−	−	+	+	−	+	+
a	−	+	+	+	−	−	−	−	−	−	−	−	−	+	−	+
o	−	+	+	+	−	−	−	−	−	−	−	+	+	+	+	+
u	−	+	+	+	−	−	−	−	−	−	−	+	−	+	+	+
ɪ	−	+	+	+	−	−	−	−	−	−	−	+	+	−	−	−
ɛ	−	+	+	+	−	−	−	−	−	−	−	−	+	−	−	−
ɑ	−	+	+	+	−	−	−	−	−	−	−	−	−	+	−	−
ɔ	−	+	+	+	−	−	−	−	−	−	−	−	+	+	+	−
ʏ	−	+	+	+	−	−	−	−	−	−	−	+	+	+	+	−
ə	−	+	+	+	−	−	−	−	−	−	−	−	−	−	−	−

independently of their frequency of occurrence. These two simulations allow us to investigate in detail the effects of using type-based and token-based training regimes.

2.1. Method

2.1.1. Materials

The phonemic representations for the words as available in CELEX (Baayen, Piepenbrock, & Gulikers, 1995), determined the phonetic features activated in the input, according to Table 1. However, we made some systematic adjustments to the CELEX phonemic representations to make them more realistic phonetically. We doubled all phonetically long vowels in stressed positions: /a/, /e/, /o/, and /ø/ in stressed syllables were substituted for [aa], [ee], [oo], and [øø], respectively (cf., Rietveld, Kerkhoffs, & Gussenhoven, 1999). Moreover, we simulated the diphthongization of /e/, /ø/, and /o/ (Ernestus, 2000), by averaging the second part of the long vowel with [j] in the case of /e/, and with [w] in the cases of /o/ and /ø/. However, when the long stressed vowels preceded an /r/, we did not average the final part of the vowel with the glide, as in this context the vowels tend more to end in a schwa.

We used all 121,529 Dutch forms present in the CELEX lexical database in a pretraining phase. The purpose of this phase was to provide the networks with some basic information about Dutch orthography before training it on the actual past-tense formation task. The orthographic output form of the word on which the network was trained was reduced to its last letter in this pre-training phase.

For the training phase itself, we used all 2957 first person singular present-tense Dutch verb forms with regular past-tenses and a frequency greater than zero in CELEX. The outputs at this phase were the last letter of the verb stem followed by the two letters that form its regular past-tense allomorph (*de* or *te*).

Finally, for testing the networks, we used the 165 existing Dutch verbs from Ernestus and Baayen (2001) (which all had CELEX frequencies greater than zero and thus also appeared in the training set), and the 145 pseudo-verbs from Ernestus and Baayen (2003).

2.1.2. Procedure

We modeled the networks using the Light Efficient Network Simulator (LENS; Rohde, 1999). Training was done using the modified momentum descent algorithm described by Rohde and Plaut (1999), using cross-entropy as the error measure on normalized outputs. For the training, we used a learning rate of 0.04, and a momentum of 0.90.

In the pre-training phase, the words from the pre-training dataset were presented to both networks in a pseudo-randomized order, each word being presented to the networks a mean number of times that was proportional to its logarithmic surface CELEX frequency. In total, the number of word tokens presented to the networks equaled the number of word types in the data set, that is, 121,529. This pre-training went on for 100 epochs: The networks were presented with 121,529 randomly chosen word tokens for 100 times. During this pre-training phase, the input and output weights to and from three of the ten hidden units were frozen, and thus were not affected by pre-training. In this way, we guaranteed that not all of the networks' memory would be used in learning Dutch orthography, leaving some space for the actual past-tense formation problem. The weights to and from the remaining seven units were adjusted on the basis of their orthographical outputs for the final letters of the words.

After the pre-training phase, the weights of all units in the hidden layer were released, allowing training to proceed in all of them during the training phase. Then, one network was trained with a type-based regime, and the other with a token-based regime. Both networks were trained to produce the last letter of the verb stem and the letters of the past-tense suffix for all the verbs in the training set. The networks received the first person singular present-tense forms of these verbs phoneme by phoneme, one at a time. The networks' weights were adjusted on the basis of their outputs for the stem-final segments and the past-tense allomorphs.

The network from Simulation 1a (token-based) was trained for eight epochs, in which examples were randomly chosen from the experimental list according to their frequencies. After eight epochs, the training was stopped because further training seemed to impoverish the network's performance (over-training), as appeared from a small test set of zero-frequency verbs from the CELEX database (which were not present in the training set or in the experimental datasets). After training, the mean square error per output unit on the training set was 0.0043 (equivalent to $\pm 6.56\%$ of each unit's correct activation value).

The network from Simulation 1b (type-based) was trained for seven epochs, after which its performance on the verbs in the small testing set seemed to drop. In this Simulation, words were presented randomly according to a uniform probability distribution (all words were on average presented an equal number of times per epoch). After training, the mean square error per output unit on the training set was 0.0045 (equivalent to $\pm 6.71\%$ of each unit's correct activation value).

Testing proceeded in the same way in both networks. We presented the verbs from the two experimental datasets, introducing the first person singular present-tense form one phoneme at a time. Once the trigger bit had been activated, we started recording the activation of the output units in that time-step and the two following steps. These three time-steps corresponded to the last

letter of the stem and the two letters of the past-tense suffix. For the first letter of the past-tense suffix, given that the output could only be 't' or 'd', it was not necessary to record the activation at the nodes representing the letters other than 't' or 'd', which was zero in all cases. Note that the activations of the 't' and 'd' nodes gave us an estimation of the probabilities of choosing between the *de* and *te* suffixes as estimated by the network.

2.2. Results and discussion

We started our evaluation by attributing *de* as the network's response when the activation of the 'd' output node was greater than the activation of the 't' output node, and *te* in the opposite cases. We compared these networks' choices with the majority choices of the participants in the experiments reported in Ernestus and Baayen (2001, 2003). Both networks showed above chance agreement with the participants according to the κ statistic for inter-rater agreement (Guggenmoos-Holzmann, 1996). The token-based model showed a coherence score with the participants' majority choices on the pseudo-words of 79% ($\kappa = 0.50$, $SE = 0.08$, $Z = 6.44$, $p < .0001$) and 78% on the existing words ($\kappa = 0.51$, $SE = 0.07$, $Z = 7.28$, $p < .0001$). The network that received a type-based training outperformed the one that received a token-based training on the pseudo-verbs. It showed a coherence score on pseudo-words of 91% ($\kappa = 0.76$, $SE = 0.08$, $Z = 9.14$, $p < .0001$) and of 78% on the existing words ($\kappa = 0.53$, $SE = 0.07$, $Z = 7.23$, $p < .0001$).

The Spearman rank correlation coefficients between the activation of the networks' 'd' output nodes and the proportion of *de* responses were very similar for the two networks, both for the pseudo-verbs ($r_s = .63$, $p < .0001$ token-based, versus $r_s = .65$, $p < .0001$ type-based), and the existing Dutch verbs ($r_s = .70$, $p < .0001$ token-based, versus $r_s = .69$, $p < .0001$ type-based), indicating that the two networks provide an equally good fit to the participants' responses. We conclude that both training regimes result in similar performances in terms of reproducing the participants' behavior on the experimental tasks, with a slight advantage on the pseudo-words for the network that was trained with a type-based training regime. Since both networks perform well on the pseudo-verbs, we can conclude that they have developed analogical behavior, like the participants did.

When we investigated whether the networks also showed the purely token-based surface frequency effects found by Ernestus and Baayen (2001) for the existing verbs, we obtained very different results. While the log frequency of a past-tense form correlated with the average number of non-standard responses produced by the participants for that form ($r_s = -.24$, $p = .0016$), this correlation only reached significance in the token-based training regime ($r_s = -.30$, $p = .0002$), and not in the type-based training regime ($r_s = -.08$, $p = .32$).

These two models have a number of limitations. The type-based training is unnatural, as actual word frequencies are not uniformly distributed. In addition, both models under performed in producing past-tenses for existing verbs: The average coherence score between a given participant's choice of past-tense allomorph and the allomorph chosen by the majority of the other participants was 90%, which is significantly above the performance of both networks. This is probably due to the small memory of the networks, in which only three hidden units bear most of the burden of past-tense formation. This substantial limitation on representational space does not allow the networks to form individual lexical representations for existing verbs, and analogical generalizations at the same time. This also explains why the type-based training regime showed an astonishing performance on pseudo-words, indicating that it succeeded in capturing the analogical effects, while it had a paradoxically lower performance on producing the past-tenses of the words on which it had been trained.

The present type and token-based models are limited also in other respects. By training the models on producing the last letter of the stem followed by the past-tense allomorph, we have implicitly provided the information that the last segment of the stem is crucial for the choice of the past-tense allomorph. This simplified the task by inducing the networks to base their choice of past-tense allomorph primarily on the last letter of the stem, without taking other properties of the verbs into account.

In addition, the input to the models during training was restricted to singular regular past-tenses. The exclusion of the irregulars decreased memory load, but oversimplified the Dutch past-tense formation problem. At the same time, the exclusion of regular plural forms from the networks' training entailed that the networks had no access to a potential natural source of information on past-tense formation. Dutch regular plural present-tense forms consist of the verb stem followed by the suffix *en*. The stem final obstruent before this suffix is not devoiced, and therefore it accurately predicts the appropriate past-tense allomorph for all forms of that particular verb (when unvoiced the allomorph is *te*, otherwise it is *de*). Given the strong degree of similarity between the singular and plural regular past-tense forms of the same verb (they only differ orthographically in the presence of the letter 'n' at the end of the plural), the information about the plural facilitates formation of the past-tense for the singular.

We conclude that a more realistic model of Dutch past-tense formation requires a larger memory, that it should produce full verbal forms, and, finally, that it should also be trained to produce plural and irregular past-tenses as well.

3. Simulation 2

There are two possibilities for a network that is required to output full verbal forms. Either the output has to be sequential, which is problematic (Stoianov, 2001), or the full form has to be output in a single time-step. Some representational paradigms have been used for single time-step output. These paradigms use either templates (e.g., Plunkett & Juola, 1999) or 'wickelfeatures' (e.g., Rumelhart & McClelland, 1986). Templates include crucial information about word structure specific to a particular language, and require unrealistic preprocessing of the inputs by alignment, etc.. They also impose explicit constraints on word length, which makes them unsuitable for our task. Wickelfeatures have been claimed to represent strings of unlimited length but they are in fact incapable of representing unambiguously all strings in a language (Prince & Pinker, 1988).

We therefore made use of the 'Accumulation of Expectations' technique (AoE; Moscoso del Prado Martín, Schreuder, & Baayen, in press). This technique is inspired by the simulations described by Elman (1990, 1993), which show that, when an SRN is trained on predicting the next letter in a sequence, using a large enough corpus for training, it develops in its hidden layer a detailed representation of the orthography of the language. The AoE for a word is the activation of the hidden units of an SRN trained to predict the next letter in a sequence of letters, summed across the presentation of each letter of the word. This vector thus gives a distributed representation of the orthographic form of the word. The AoE technique allows us to create vectors representing the orthographical forms of Dutch words (and pseudo-words) of any length, implicitly incorporating the generalizations of the orthographical system of Dutch.

In contrast to Simulation 1, the training input contained all first and third person present- and past-tense verbal forms, regulars or irregulars. We used a past-tense formation SRN very similar to the ones used in Simulation 1. The phonetic input layer remained exactly the same as in Simulation 1, using the same feature-based representations of the input together with the trigger bit. We extended the memory to 200 units in the hidden and context layers, as the model has to learn mappings for full words, including irregulars. The output layer consisted of 50 units that represent the AoE of the input words.

Fig. 2 provides an outline of the complete model that was used in the present simulation. The model consisted of two parts: An AoE network, that was used to create flat orthographic representations for the words and pseudo-words (upper part of the figure), and a past-tense formation network similar to the one employed in Simulation 1. The representations created by the AoE network were employed for evaluating the outputs of

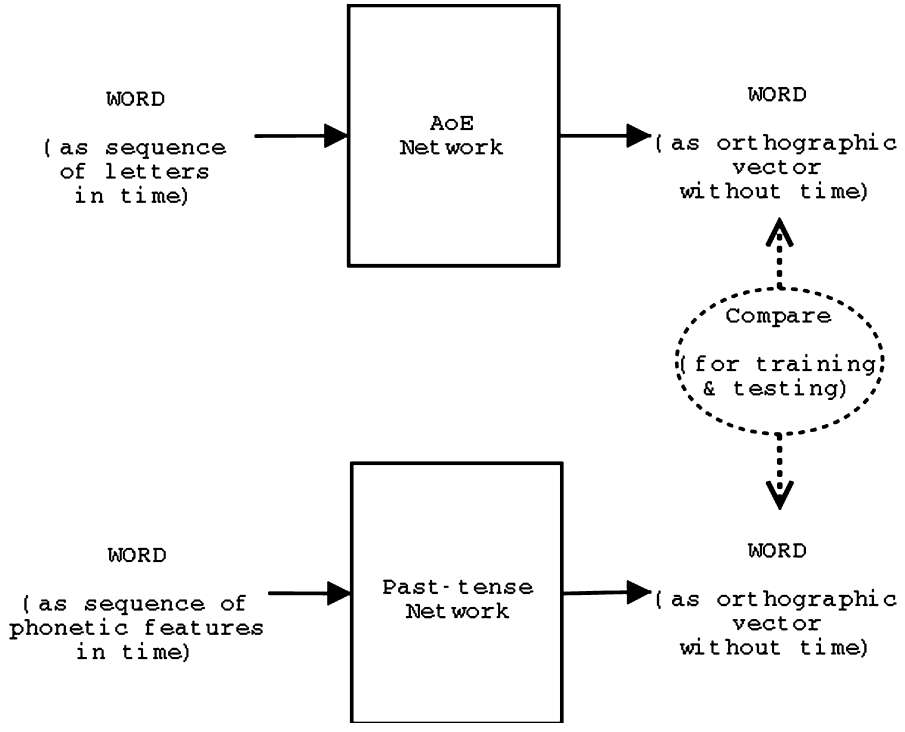

Fig. 2. Outline of the complete model used in Simulation 2. The upper half of the diagram represents the AoE network that created orthographic vectors corresponding to sequences of letters. These vectors were used for training and evaluating the performance of the past-tense network depicted in the lower part of the diagram.

the past-tense formation network, that is, for training, testing, and interpreting the outputs of this network.

3.1. Method

3.1.1. Materials

The AoE network was an SRN with 50 units in its hidden and context layers. The input and output layers consisted of 26 units, each of these corresponding to one letter of the Dutch alphabet. We trained this network on predicting the next letter, using all 297,690 Dutch words in CELEX as training set. The words were presented letter by letter, and at each time-step the AoE network was trained on predicting the next letter in the word. After training the AoE network, we ran through it all Dutch words in the CELEX database, and we accumulated the activation vectors produced in the hidden layer after each phoneme. In this way, we obtain a vector of 50 numbers for each word, with values in the interval [0.0, 1.0]. The past-tense formation network in this simulation was trained to produce these same vectors as its output orthographic representations. A more detailed description of this AoE network can be found in Moscoso del Prado Martín et al. (in press).

For the pre-training phase of the past-tense formation network, we used again all 121,529 Dutch words from the CELEX lexical database with frequencies higher than zero. The phonological coding of the input words was the same as in Simulation 1. Outputs were coded using the AoE technique.

For the training phase of the past-tense formation network, we selected all 10,750 present- and past-tense, regular and irregular, Dutch first and third person verbal forms that appear in the CELEX database with a frequency higher than zero. The input consisted of the phonological form corresponding to the present-tense forms. The coding of the inputs was done according to the method described in Simulation 1, with an additional bit that was set to zero at the last phoneme when an identity mapping was the required task, in which case, the model performs a pure 'dictation' task. If the required output was the corresponding past-tense form, the bit was set to one.

Again, for testing the network, we used the 145 pseudo-verbs from Ernestus and Baayen (2003) and the 165 existing Dutch verbs from Ernestus and Baayen (2001). Input and output were coded in the same manner as in the training phase.

3.1.2. Procedure

We modeled the network using LENS. Training was done using the modified momentum descent algorithm (Rohde & Plaut, 1999), this time using cosine as our error measure. In all training phases, we used a learning rate of 0.04 and a momentum of 0.90.

The network first went through a pre-training phase of 100 epochs. This phase was identical to the one in Simulation 1, except that the network was trained to produce the full orthographic forms, instead of just the last letter. Each item was presented a number of times proportional to its logarithmic frequency. The input and output weights to and from 100 of 200 hidden units were frozen, and thus were not affected by pre-training. Error was back-propagated after the presentation of the last phoneme of each word.

After the identity mapping pre-training phase, the weights of all units in the hidden layer were released, allowing training to proceed throughout the network. Training items were presented to the network in a random order a number of times that was directly proportional to the frequency of occurrence of the output form (present or past-tense). In this way we simulated the frequencies of production of present and past-tense forms. The network was trained for 2000 epochs. Training was stopped at this point because further training did not seem to improve performance, which was tested on the same set of zero-frequency verbs from Simulation 1. After training, the network's average cosine error on the training set was 0.0343 ($\pm$0.0254).

For testing, we presented the verbs from both experimental datasets, introducing the first person singular present-tense form (for both the pseudo-verbs and the existing verbs) one phoneme at a time. Once the output triggering bit had been activated, we recorded the activation of the output units in that time-step.

3.2. Results and discussion

Once more, we evaluated our model by comparing the past-tense forms chosen by the majority of the participants with the preferred output of the model. Since the output of this network was a distributed AoE representation, instead of a localistic one, determining the preferred output of the network was more complicated than in Simulation 1. Using the AoE network described above, we created 50-element vectors for the two possible orthographic forms of the regular past-tense (ending in *te* or *de*) for each of the experimental items. For example, given the experimental stimulus [kɛit], the AoE network produced 50-element vectors for the possible output strings *keidde* and *keitte*. Additionally, to investigate the effects of irregularization, we also created vectors for two irregular forms for each item, one ending in a voiced obstruent and the other ending in a unvoiced obstruent. The selection of these irregular past-tenses for the experimental items (that were either regular or nonexisting verbs) was done by choosing the most likely vowel change pattern for the final vowel and consonant clusters of the verb (or pseudo-verb) according to the majority of existing irregular verbs in CELEX. For instance, for the experimental item [kɛit], we created AoE

vectors corresponding to the orthographical forms *keet* and *keed*, because the majority of existing irregular verbs that contain the diphthong /ɛi/ undergo the /ɛi/ to /eː/ vowel change. We calculated the cosine distance between the output of the past-tense formation network for each experimental item, and the corresponding regular past-tenses ending in *de* and *te* as well as the two created irregular forms (which in most cases do not correspond to any existing Dutch word). Out of these four options, the form with the vector that had the smallest distance to the network's output was selected as the network's preferred choice.

In order to assess whether the network was indeed producing as its output one of these four possibilities, we also calculated the cosine distance between the output vector produced by the network for each experimental item (words and pseudo-words), and the orthographical vectors of all the 30,740 word forms in the CELEX database with a surface frequency of at least one occurrence per million. The cosine distance between the network's output and the closest form among the four predefined choices was smaller (or equal in cases of existing verbs) than the distance to the closest existing word from CELEX in 85% of the existing words and 63% of the pseudo-words. The outputs for the remaining 15% of words and 37% of pseudo-words were marked by deviant spellings, often caused by ambiguity in Dutch phoneme to grapheme mappings (e.g., the diphthongs /ɑu/ and /ɛi/ can be respectively spelled as *au* or *ou*, and *ei* or *ij* in Dutch). For some pseudo-words, the network produced intermediate representations between the different possibilities because these cases can only be resolved by memorizing the correct spelling for each word. Other errors involved small 'misperceptions,' that is, pseudo-words were mistaken for similar-sounding existing words. For instance, when presented with [bɑus], the network produced the existing verb *bouwde* (''built''), instead of inflected versions of *baus* or *bous*. Finally, for a small proportion of the verbs, the network produced 'impossible' morphological forms, such as attaching *te* after a voiced consonant (e.g., *bembte*), or *de* after an unvoiced one (e.g., *daantde*). Interestingly, Ernestus and Baayen (2001) reported that participants also produced this type of errors.

The model chose an irregular past-tense for 40 experimental items by changing the vowel without affixation of a past-tense allomorph. For instance, it created *keed* instead of *keidde* or *keitte*, as the past-tense form of [kɛit], in analogy with existing Dutch verbs. Interestingly, although the training was token-based, and the irregular past-tenses were consequently more frequent than the regular ones, the network's preferred past-tense was irregular for only 25 out of 145 (14%) pseudo-verbs and 15 out of the 165 (9%) existing Dutch regular verbs, in line with percentages of irregularization reported in the literature (e.g., Albright & Hayes, 2003 report 18.5% irregularization in two experiments on English pseudo-verbs). Some of the irregularizations are in fact correct as they correspond to existing homophonic irregular verbs (e.g., the network produced *liet*, the past-tense form of the verb *laten*, ''to let'', instead of *laadde*, the past-tense form of *laden*, ''to load'', upon presentation of [laːt]). Irregularization occurred more often when the irregularized form was also an existing (usually unrelated) word. In Ernestus and Baayen's experiments participants were explicitly instructed to produce regular past-tense forms in all cases. We therefore excluded these 40 irregularized items from the following analyses.

The comparison of the network's preferred choices ending in *te* or *de* with the participants' majority choices gave significantly above chance coherence scores of 68% for the pseudo-verbs ($\kappa = 0.33$, $SE = 0.07$, $Z = 4.5$, $p < .0001$) and 85% for the existing Dutch verbs ($\kappa = 0.70$, $SE = 0.08$, $Z = 8.6$, $p < .0001$). Interestingly, the participants in Ernestus and Baayen's experiments showed similar average coherence scores with each other: 74% (±5%) for the pseudo-verbs, and 90% (±5%) for the existing verbs. The coherence scores of the participants did not differ significantly from the coherence scores shown by the network, neither for the pseudo-words ($Z = -1.20$, $p = .1151$), nor for the existing verbs ($Z = -1.00$, $p = .1587$).

We also calculated the correlations between the logits for the participants' responses, that is, the logarithmic ratio between the number of *de* responses and the number of *te* responses for a given verb, and an estimation of the logit values for the network outputs. We estimated the network logits using $\hat{L} = \log(d_{te}/d_{de})$, with d_{de} being the cosine distance between the *de* form and the network output, and d_{te} being the cosine distance between the *te* form and the network output. The Spearman rank correlations between the logits of the number of *de* and *te* responses produced by the participants, and the L values were $r_s = .50$ for the pseudo-verbs and $r_s = .76$ for the existing Dutch verbs ($p < .0001$ in both cases). The correlation between the participants' logits and our estimated network logits is illustrated in Fig. 3. The regression line shows that the network's outputs replicate the participants' behavior: A linear increase of the network's logits is proportional to the linear increase of the participants' logits.

The correlation between the network's estimated logits and the participants' logits suggests that our network is showing the type-based analogical effects described by Ernestus and Baayen (2001, 2003). Fig. 4 provides a more detailed account of these analogical effects. The figure compares the estimated logit values produced by the network, with the logits of the participant responses to different groups of pseudo-words classified by type of final obstruent of the pseudo-verb (upper panel), type of segment preceding the final obstruent of the pseudo-verb (middle panel), and the

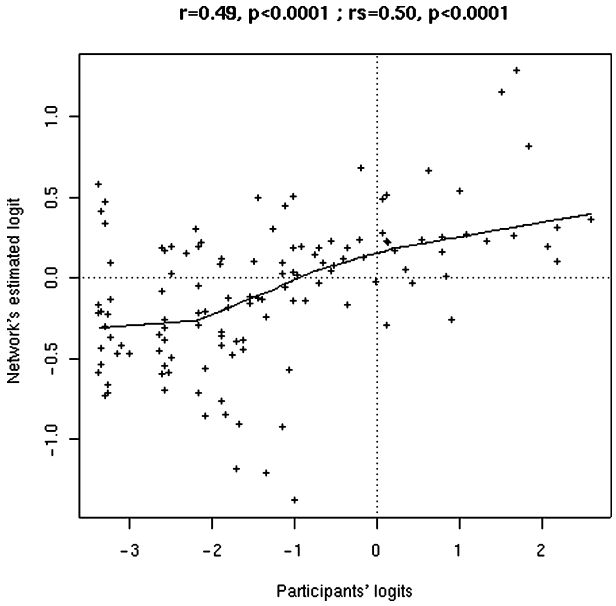

Fig. 3. Comparison between the logit values of the proportion of *de* and *te* responses to pseudo-verbs by the participants in Ernestus and Baayen (2003), with the estimated logit values of the responses of the network in Simulation 2. The line represents a non-parametric regression (Cleveland, 1979).

quantity of the final vowel (lower panel). Note that the network replicates accurately the patterns showed by the participants, with only small differences of scaling.

The network's error for each verb was estimated by the absolute value of the estimated network logit ($|\hat{L}|$). This estimate represents how confident the network was in its choice of past-tense. High values should correspond to few errors produced by the participants. The Spearman correlation between this error estimate and the number of errors produced by the participants for the existing verbs was $r_s = -.44$ ($p < .0001$). This correlation shows that the network was replicating not only the participants' preferred choices, but also the participants' certainty about a particular choice. Finally, the network showed higher confidence for the more frequent verbs ($r_s = -.55$; $p < .0001$), thus, replicating the token-based frequency effect in the participants' responses.

To explore the network's performance on irregular verbs, we selected the 153 Dutch monomorphemic irregular verbs from the CELEX lexical database (excluding the verbs that could have more than one past-tense form), and we ran them through the network in their first person singular present-tense form, producing their past-tenses. We compared the output of the net-

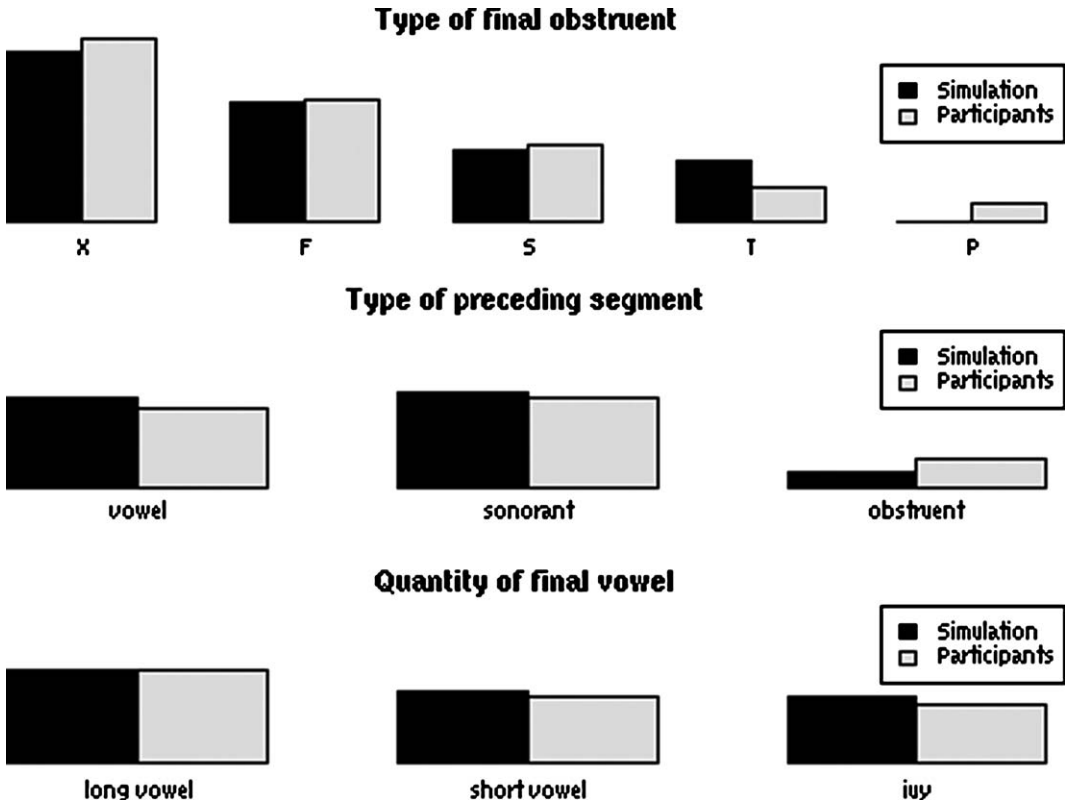

Fig. 4. Comparison between the standardized logit values of the proportion of *de* and the *te* responses to pseudo-verbs by the participants in Ernestus and Baayen (2003), with the standardized logit values of the responses of the network in Simulation 2. The pseudo-verbs are classified by type of final obstruent (upper panel), type of segment preceding the final obstruent (middle panel), and quantity of the final vowel (lower panel).

work with the AoE vectors corresponding to the correct irregular past-tense form, and with the two possible regularizations (using the *te* or *de* allomorphs) for each verb. In 88% of the cases, the distance between the output of the network and the correct irregular past-tense form was smaller than the distance to any of the two possible regularizations. Additionally, we found that the network's output distance to the chosen past-tense form, was in 85% of the cases smaller or equal than the distance to the closest existing Dutch word with a frequency of at least one occurrence per million.

This new model succeeds in capturing type-based analogical effects and token-based frequency effects in a single system with a token-based training regime. This shows that the combined presence of token-based and type-based effects does not necessarily imply two different mechanisms. The network's performance is remarkable given that, in this simulation, the task was much more complicated than in Simulation 1, requiring the learning of the full Dutch phoneme to grapheme mappings and the whole past-tense formation system, including irregulars and plurals, with a still very limited amount of memory.

4. General discussion

In this paper, we discussed three neural networks modeling past-tense formation in Dutch. The first two networks dealt only with regular past-tense formation in the singular, one receiving a type-based training, the other one a token-based training. The inputs for both models consisted of featural phonetic representations of verbal stems simulating auditory input. As outputs, the models produced the final letters of the verbal stems followed by their past-tense allomorphs. The token-based model replicated the type and token-based effects reported by Ernestus and Baayen (2001, 2003). The model that received a type-based training regime matched the experimental results for the pseudo-verbs in more detail, but it failed to replicate the token-based frequency effects for the existing verbs. In fact, both of the models showed a relatively low performance on modelling the participants' responses to existing verbs. We concluded that accurate modeling of past-tense formation is only possible when a not too small memory is available, allowing both for the storage of individual items, and for the formulation of generalizations.

The third model received only a token-based training regime. Its memory was much larger, and the training set contained both regular and irregular, and both singular and plural past-tense forms. Furthermore, the model not only chose a past-tense allomorph, but also provided full orthographic forms. This model displays the type-based analogical effect for both the existing verbs and the pseudo-verbs, together with the token-based frequency effect for the existing verbs, closely replicating humans' responses to those same items. We conclude that type and token-based effects in morphological processing do not necessarily imply the existence of separate processing mechanisms. Type-based analogical effects can arise as a consequence of uncertainty in token-based probability distributions as it was proposed by Moscoso del Prado Martín Kostć et al. (in press).

Our system contributes to the ongoing debate on single and dual route models for regular and irregular past-tense formation (for reviews see McClelland & Patterson, 2002a, 2002b; Pinker & Ullman, 2002a, 2002b) by showing that both regulars and irregulars can be captured by a simple model trained on a realistic amount of different verbs according to the best available estimates of their frequencies. Our system produced regular past-tense forms for the great majority of pseudo-verbs, showing a default rule for past-tense formation. At the same time, it produced irregular past-tenses for existing irregular verbs, and for only a few pseudo-verbs with strong analogical support for the irregular form. This shows that analogical processing does not exclude rule-like generalizations. Default rules, in the sense of Pinker (1999), can arise in an exemplar-based connectionist system.

As far as we know, this is the first model of past-tense formation that covers all verbs of a language, irrespective of their length, regularity, or morphological complexity. Pinker and Ullman (2002a, 2002b) argue that previous connectionist models of past-tense formation (e.g., Plunkett & Juola, 1999; Plunkett & Marchman, 1993; Rumelhart & McClelland, 1986) only succeed in this task because a great deal of linguistic knowledge was built into their systems, thus making them more similar to a symbolic model. In contrast, our third simulation succeeded in this task using only phonetic representations. It acquired the remaining structural knowledge by statistical generalizations over the phonological and orthographical sequences present in words, without making use of any implicit symbolic processing mechanism.

References

Albright, A., & Hayes, B. (2003). Rules vs. analogy in English past tenses: A computational/experimental study. *Cognition, 90*(2), 119–161.

Baayen, R. H., Dijkstra, T., & Schreuder, R. (1997). Singulars and plurals in Dutch: Evidence for a parallel dual route model. *Journal of memory and language, 37*, 94–117.

Baayen, R. H., Piepenbrock, R., & Gulikers, L. (1995). *The CELEX lexical database (CD-ROM)*. Philadelphia, PA: Linguistic Data Consortium, University of Pennsylvania.

Baayen, R. H., Schreuder, R., De Jong, N. H., & Krott, A. (2002). Dutch inflection: The rules that prove the exception. In S. Nooteboom, F. Weerman, & F. Wijnen (Eds.), *Storage and*

computation in the language faculty (pp. 61–92). Dordrecht: Kluwer Academic Publishers.

Booij, G. E. (1995). *The phonology of Dutch*. Oxford: Clarendon Press.

Bybee, J. L. (1995). Diachronic and typological properties of morphology and their implications for representation. In L. B. Feldman (Ed.), *Morphological aspects of language processing* (pp. 225–246). Hillsdale, NJ: Lawrence Erlbaum Associates.

Bybee, J. L. (2001). *Phonology and language use*. Cambridge: Cambridge University Press.

Clahsen, H. (1999). Lexical entries and rules of language: A multidisciplinary study of German inflection. *Behavioral and brain sciences, 22*, 991–1060.

Cleveland, W. S. (1979). Robust locally weighted regression and smoothing scatterplots. *Journal of the American statistical association, 74*, 829–836.

Elman, J. L. (1990). Finding structure in time. *Cognitive science, 14*, 179–211.

Elman, J. L. (1993). Learning and development in neural networks: The importance of starting small. *Cognition, 48*, 71–99.

Ernestus, M. (2000). *Voice assimilation and segment reduction in casual Dutch, a corpus-based study of the phonology–phonetics interface*. Utrecht: LOT.

Ernestus, M., & Baayen, R. H. (2001). Choosing between the Dutch past-tense suffixes *-te* and *-de*. In T. van der Wouden & H. de Hoop (Eds.), *Linguistics in the Netherlands 2001* (pp. 81–93). Amsterdam: Benjamins.

Ernestus, M., & Baayen, R. H. (2003). Predicting the unpredictable: Interpreting neutralized segments in Dutch. *Language, 79*(1), 5–38.

Guggenmoos-Holzmann, I. (1996). The meaning of kappa: probabilistic concepts of reliability and validity revisited. *Journal of clinical epidemiology, 49*(7), 775–782.

McClelland, J. L., & Patterson, K. (2002a). Rules or connections in past-tense inflections: what does the evidence rule out? *Trends in the cognitive sciences, 6*(11), 465–472.

McClelland, J. L., & Patterson, K. (2002b). 'Words or rules' cannot exploit the regularity in exceptions: Reply to Pinker and Ullman. *Trends in the cognitive sciences, 6*(11), 464–465.

Moscoso del Prado Martín, F., Kostić, A., & Baayen, R. H. (in press). Putting the bits together: An information theoretical perspective on morphological processing. *Cognition*.

Moscoso del Prado Martín, F., Schreuder, R., & Baayen, R. H. (in press). Using the structure found in time: Building real-scale orthographic and phonetic representations by accumulation of expectations. In H. Bowman & C. Labiouse (Eds.), *Connectionist models of cognition and emotion: proceedings of the VIII international workshop on neural computation and psychology*. Singapore: World Scientific.

Pinker, S. (1999). *Words and rules: The ingredients of language*. London: Weidenfeld and Nicolson.

Pinker, S., & Prince, A. (1994). Regular and irregular morphology and the psychological status of rules of grammar. In S. Lima, R. Corrigan, & G. Iverson (Eds.), *The reality of linguistic rules*. Amsterdam: John Benjamins.

Pinker, S., & Ullman, M. (2002a). Combination and structure, not gradedness, is the issue: Reply to McClelland and Patterson. *Trends in the cognitive sciences, 6*(11), 472–474.

Pinker, S., & Ullman, M. (2002b). The past and future of the past tense. *Trends in the cognitive sciences, 6*(11), 456–462.

Plunkett, K., & Juola, P. (1999). A connectionist model of English past tense and plural morphology. *Cognitive science, 23*(4), 463–490.

Plunkett, K., & Marchman, V. (1993). From rote learning to system building: Acquiring verb morphology in children and connectionist nets. *Cognition, 48*, 21–69.

Prince, A., & Pinker, S. (1988). Wickelphone ambiguity. *Cognition, 30*, 189–190.

Rietveld, T., Kerkhoffs, J., & Gussenhoven, C. (1999). Prosodic structure and vowel duration in Dutch. in Ohala, J., Hasegawa, Y., Ohala, M., Granville, D., & Baily, A. (Eds.), *Proceedings of the 14th International Congress of Phonetic Sciences, San Francisco, 1–7 August 1999*, Linguistic Department, University of California, Berkeley, pp. 463–466.

Rohde, D.L.T. (1999). LENS: The light, efficient network simulator. Technical Report CMU-CS-99-164, Pittsburg, PA: Carnegie Mellon University.

Rohde, D. L. T., & Plaut, D. C. (1999). Language acquisition in the absence of explicit negative evidence: How important is starting small? *Cognition, 72*(1), 67–109.

Rumelhart, D. E., & McClelland, J. L. (1986). On learning the past tenses of English verbs. *Parallel distributed processing. Explorations in the microstructure of cognition* (Vol. 2, pp. 216–271). Cambridge, MA: The MIT Press.

Schreuder, R., De Jong, N. H., Krott, A., & Baayen, R. H. (1999). Rules and rote: Beyond the linguistic either-or fallacy. *Behavioral and brain sciences, 22*, 1038–1039.

Stoianov, I.P. (2001). *Connectionist Lexical Processing*, Ph.D thesis, Rijksuniversiteit Groningen, Groningen, The Netherlands.