COGNITIVE SCIENCE
A Multidisciplinary Journal

Cognitive Science 37 (2013) 1193–1227

ISSN: 0364-0213 print / 1551-6709 online
DOI: 10.1111/cogs.12072

Birth of an Abstraction: A Dynamical Systems Account of the Discovery of an Elsewhere Principle in a Category Learning Task

Whitney Tabor,[a] Pyeong W. Cho,[a] Harry Dankowicz[b]

[a] *Haskins Laboratories, University of Connecticut*
[b] *Department of Mechanical Science and Engineering, University of Illinois at Urbana-Champaign*

Received 17 January 2012; received in revised form 30 December 2012; accepted 8 January 2013

Abstract

Human participants and recurrent ("connectionist") neural networks were both trained on a categorization system abstractly similar to natural language systems involving irregular ("strong") classes and a default class. Both the humans and the networks exhibited staged learning and a generalization pattern reminiscent of the Elsewhere Condition (Kiparsky, 1973). Previous connectionist accounts of related phenomena have often been vague about the nature of the networks' encoding systems. We analyzed our network using dynamical systems theory, revealing topological and geometric properties that can be directly compared with the mechanisms of non-connectionist, rule-based accounts. The results reveal that the networks "contain" structures related to mechanisms posited by rule-based models, partly vindicating the insights of these models. On the other hand, they support the one mechanism (OM), as opposed to the more than one mechanism (MOM), view of symbolic abstraction by showing how the appearance of MOM behavior can arise emergently from one underlying set of principles. The key new contribution of this study is to show that dynamical systems theory can allow us to explicitly characterize the relationship between the two perspectives in implemented models.

Keywords: Dynamical systems theory; Elsewhere Condition; Rule learning; Default Categorization; Recurrent neural networks; Connectionist (neural network) modeling; Abstraction; Emergence

1. Introduction

In a number of situations, learners go from a stage of being rooted in particulars to a stage of embodying an abstract principle that goes beyond the particulars they have

Correspondence should be sent to Whitney Tabor, Department of Psychology, University of Connecticut, 406 Babbidge Road U-1020, Storrs, CT 06269. E-mail: whitney.tabor@uconn.edu

observed. A much discussed case is the "U-shaped" learning of the past tense of English verbs: Children transition from what appears to be a rote strategy (each past tense memorized) to what appears to be a rule-based strategy (add *–ed* to stem to form past). They first inflect a few high-frequency verbs, both regular (e.g., *walk/walked*) and irregular (or "strong" verbs; e.g., *give/gave*) correctly; then, for a short period of time they tend to over-apply the *–ed* rule, even using it with verbs that they originally inflected correctly (e.g., *gived*); eventually, they correct these mistakes (Berko, 1958; Cazden, 1968; Maratsos, 2000; Marcus et al., 1992; see McClelland & Patterson, 2002; Pinker & Ullman, 2002a,b for discussion). We describe a category learning experiment that produces a distinct, but related transition to abstraction in humans in a single laboratory session. We also present a recurrent neural network model of the human behavior. We then describe a dynamical systems analysis of the network that makes it clear how the model exhibits both particular behavior and abstract behavior. Our results favor the one mechanism (OM) view of how cognition exhibits both specific and general behavior (e.g., Hare, Elman, & Daugherty, 1995; Laakso & Calvo, 2011; Plunkett & Nakisa, 1997; Rumelhart & McClelland, 1986), although they suggest that care is needed in clarifying what "one mechanism" means because it is also reasonable to claim that multiple mechanisms arise emergently in the system we describe (McClelland et al., 2010). Although we agree with the broad OM claims of many connectionist researchers, we argue that the jury is still out on whether connectionist models employ one mechanism or more than one mechanism (MOM; see Endress & Bonatti, 2007; Peña, Bonatti, Nespor, & Mehler, 2002; Pinker & Prince, 1988), as, in most cases, the principles of the network's operation are not well understood. We claim that dynamical systems tools applied to recurrent neural networks can help clarify the situation by revealing principles of organization in the learning models.

To achieve lucid analysis, we designed a very simple category learning task in which the distinction between strong and default categories plays a central role. We therefore turn now to a review of relevant category learning models.

1.1. Category learning models

The traditional deductive approach to category learning assumes that the learner begins with a set of hypotheses and then rules out hypotheses as evidence accrues (Gold, 1967; see Mitchell, 1997). Connectionist (neural network) models have been offered as an alternative (Bartos, 2002; Kruschke, 1992; Mareschal, French, & Quinn, 2000; Regier, 1996): They seem to offer a generic mechanism for inducing structured hypotheses from data. However, the organizational structure of the trained models is often not clearly understood at a higher level of description than the description of the activation and weight dynamics, so it is not clear that they constitute an alternative to hypothesis elimination.

Recently, Bayesian models have provided a more nuanced theory (e.g., Kruschke, 2008; Xu & Tenenbaum, 2007): Bayesian models express states of the learner as probability distributions over hypotheses so they are not limited to the all-or-nothing treatment of the hypothesis elimination view, and they offer a principled approach to updating the probabilities on the basis of evidence. Bayesian models are easier to understand at the level of structured hypotheses than many connectionist models because they take the structured hypotheses as a starting point. But most Bayesian models depend for their success on starting with a restrictive initial hypothesis space (Xu & Tenenbaum, 2007), and they do not address the question of how the restricted space is established. Hierarchical empirical Bayes (e.g., Friston, 2003), employing layered recurrently connected networks, is promising, but does not (yet) give us insight into what kinds of structures are induced. We suggest that by applying dynamical systems analysis to recurrent neural networks, we can gain insight into both how induction occurs and what is induced.

Several connectionist, category learning models propose parallel rule activation–based and implicit (procedural learning–based) systems that compete to categorize the environment (Ashby, Alfonso-Reese, Turken, & Waldron, 1998; Cowell & French, 2011). These are helpful with respect to the goal (which we share) of gaining high-level insight into learning neural systems by relating them to symbolic descriptions. However, we find that in recurrent connectionist networks, homologs of symbolic structures arise emergently, even in the absence of a rule-selection network running in parallel. We seek to understand these emergent properties directly, by analyzing the network dynamics.

We turn now to a brief review of default categorization in natural language to motivate the design of our simple artificial learning task.

1.2. A linguistic view of default categorization

Often, a language will have several methods of indicating a grammatical feature (e.g., for English plural add *–ren (with vowel change), −0, −s*, etc.). Many such methods are correlated with particular features (e.g., phonological, semantic, or lexically specific) of the words receiving the feature. But at least one method will typically be a "default": It applies in all the cases where nothing else does (e.g., the *−s* plural in English). Kiparsky (1973) argued that the rule-ordering patterns of many phonological systems can be explained by a grammatical principle, the Elsewhere Condition: the Elsewhere Condition says, in essence, that in cases where two rules are both applicable but the domain of application of one of the rules is a proper subset of the domain of the other, the rule with the more specific domain takes precedence.[1] When multiple specific rules are defined on subsets of the domain of application of the same general rule, a default classification situation arises: Apply a specific rule if relevant; otherwise apply the general rule.

1.3. Evidence from language use

A number of investigations of language use have bolstered the view that there are systematic differences between regular and irregular inflections (see Pinker, 1999 for review). Bybee (1985) found that among three Old English verb classes, low-frequency forms tended to regularize over time, whereas high-frequency irregulars tended to persist. Bybee and Moder (1983) found that the likelihood of adding an irregular inflection to a nonce verb increased as a function of its phonological similarity to real verbs that take the inflection; by contrast, no such correlation with phonological properties governed participants' tendency to add regular inflections (see also Kim, Marcus, Pinker, Hollander, & Coppola, 1994; Kim, Pinker, Prince, & Prasada, 1991; Prasada & Pinker, 1993). Other studies find evidence for dissociation between irregular and regular in disordered populations (e.g., Clahsen & Almazan, 1998; Marslen-Wilson & Tyler, 1997; Ullman et al., 1997; Ullman & Gopnik, 1999) and in the brain (Beretta et al., 2003; Indefrey et al., 1997; Jaeger et al., 1996; Penke, Janssen, & Krause, 1999; Ullman et al., 1997). By and large, the findings indicate that irregulars are strongly associated with graded effects and regulars are less so, but the findings are not perfectly categorical (see Albright & Hayes, 2003; Cortese, Balota, Sergent-Marshall, Buckner, & Gold, 2006; Penke et al., 1999; Stemberger & MacWhinney, 1986).

1.4. The symbolist/connectionist debate

All these studies are motivated by the symbolist/connectionist debate about whether one or two mental mechanisms are involved in the encoding of regular and irregular inflections (McClelland & Patterson, 2002; Pinker & Ullman, 2002a,b). In support of the one mechanism view, Rumelhart and McClelland (1986) described a two-layer, feedforward connectionist network that exhibited the above-described U-shaped learning pattern when exposed to data on English verbs. Pinker and Prince (1988) critiqued Rumelhart and McClelland's model, noting, among other things, that the connectionist model was highly sensitive to the relative frequencies of its training patterns, and that the treatment of *–ed* as a default option may have stemmed from the fact that Rumelhart and McClelland flooded their model with irregular verbs just before it started overregularizing (p. 138). There are languages with morphological marking systems in which the default class (defined, for example, as the class to which novel items are assigned) is not the most frequent class (e.g., Old English noun classes, German and Arabic noun plurals). Hare et al. (1995) and Plunkett and Nakisa (1997) responded to these critiques by employing three-layer, feedforward networks trained on data resembling Old English and Arabic noun systems, respectively. They found that the networks exhibited appropriate structural distinctions between strong classes and default classes even though the default-class exemplars were not more frequent (see also Plunkett & Juola, 1999).

These studies bolster the claim that the connectionist models capture the empirical data. These connectionist results are often taken as indicating how a system might exhibit behaviors like those traditionally used to motivate rules without actually using rules. A difficulty with these claims is that it is often hard to understand why the networks do what they do or to identify general principles of their behaviors. Toward the end of their article, Hare et al. offer a tantalizing diagram that takes a step toward addressing these questions (reproduced below in Fig. 1). They suggest that the network's input space is organized into a wide, flat plane with bowls carved into various regions of the plane. Inputs that land in the bowls correspond to strong classes, whereas inputs that land on the remainder of the plane receive the default classification. The concept of a set complement[2] is a key idea underlying this diagram's implementation of the Elsewhere Condition: The default region is the

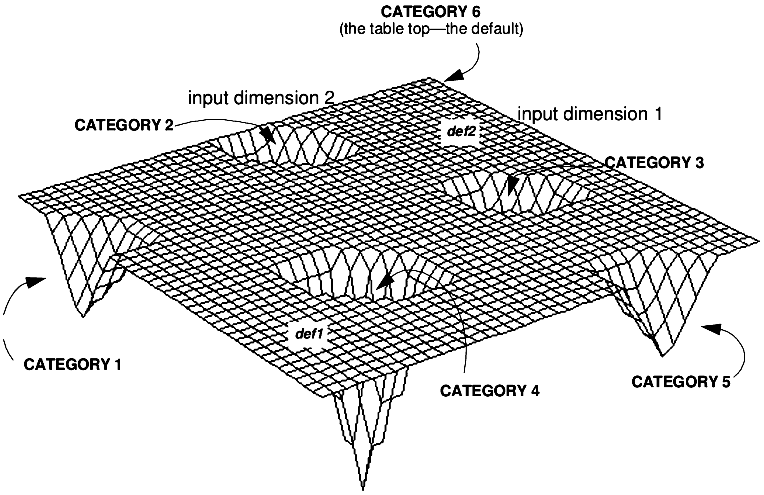


Fig. 1. Hare et al.'s (1995) Fig. 5. Caption from Hare et al.: "Hypothetical input landscape. Forms are defined in terms of input characteristics in two dimensions (x and y axes); their proximity to one of the five basins in the landscape determines their categorization. The default category is the flat surface of the "table top." Two items, def1 and def2 are also shown. Although these forms are each closer to another category than to each other, they fall outside the five basins of attraction which define the non-default categories. Since they are on the table top, which defines the "elsewhere condition," they are classified as belonging to category 6 (the default)." (figure reproduced with permission)

complement of the union of the strong regions. This diagram is appealing because it suggests a geometric form corresponding to the notion of strong versus default, thus giving some indication of how the network account might differ from a symbolic account. On the other hand, the diagram is fictional—it is not generated from a model, and its relation to the model Hare et al. examined is not specified. We think the figure is very helpful, and we offer a way of formalizing its underlying concept in a model.

1.5. Overview

Section 2 describes our category learning task. Section 3 reports the results of the network modeling. Section 4 reports the human experiment results. Section 5 provides a dynamical systems analysis of the network's behavior. This analysis supports an assessment, in Section 6, of how the network produces staged learning and how its encoding principles are related to several mechanisms employed by symbolic models.

2. Task

We trained recurrent neural networks and human participants to assign unfamiliar "animals" to classes. The classes were structured in such a way that one class abstractly resembled the default classes observed in natural languages. Table 1 gives a bit-vector portrayal of the category structure. There are four classes analogous to natural language strong classes, and one class analogous to a natural language default class. Each feature vector had six dimensions. Within the first four dimensions, each strong class was associated with a single pattern, whereas the default class was associated with four different patterns.

As we noted above, there has been much debate about whether a frequency contrast between default and strong classes is needed to make a connectionist network exhibit appropriate distinctions between regular and irregular behaviors. We included two additional features in the category structure to counterbalance the frequencies of the classes. These "additional" features took on all possible bit values for each strong class (00, 01, 10, and 11), yielding four members of each strong class. The four additional feature patterns were paired with the default class as shown in the column "Additional Features" of Table 1. The pairing of additional features with default-class items was chosen so as to enhance the heterogeneity of the default class: We suspected that humans would be especially sensitive to the number of features present on an animal so we distributed the "additional features" so as to create a range of feature counts in the default class. Under this scheme, there were also four members of the default class.[3]

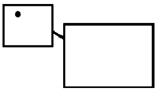
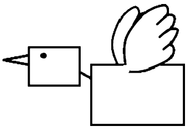
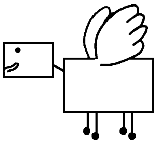
Following a common method in category learning studies, for the human experiment, we created pictures of "animals" by attaching features to different parts of a simple body (Table 2, Row 1). There were six possible features (antenna, legs, mouth, nose, tail, and wings). We mapped each picture to a six-dimensional vector by associating each location with a dimension and assigning 0 if the feature was absent and 1 if the feature was present (Table 2, Row 2).

As we noted above, the idea is that the bit vectors/pictures are analogous to different noun stems that have different phonological, semantic, etc., features and that the classes are analogous to different inflectional changes. A person learning a language must learn to choose the appropriate inflectional method for each word they encounter. This means our task has a rudimentary temporal sequencing structure analogous to the structure that children listening to native speakers receive in suffixal inflection: First the stem is heard, then the inflection is heard. For ease of analysis, we wanted to keep the presentation

Table 1
The category structure used in both the network training and the human learning experiment

Pattern No.	Core Features	Additional Features	Class
1–4	1100	00, 01, 10, 11	1 (Strong)
5–8	1010	00, 01, 10, 11	2 (Strong)
9–12	0101	00, 01, 10, 11	3 (Strong)
13–16	0010	00, 01, 10, 11	4 (Strong)
17	1000	01	5 (Default)
18	0100	11	5 (Default)
19	1110	10	5 (Default)
20	0011	00	5 (Default)

Table 2
Examples of animals created by combining features

	Base Animal	A Two-Feature Animal	A Three-Feature Animal
Human stimuli			
Network stimuli	000000	000101	011001

structure at this level of simplicity in the network models (rather than introducing a temporal sequence within the input, as in speech stimuli, or considering stem change inflections) and therefore we used visual features, presented all at once, in the human learning experiment, and bit coded features, also presented all at once, in the network learning experiments. Although human children rarely receive explicit feedback telling them that a linguistic form that they used was incorrect (Brown & Hanlon, 1970), if we assume that learners learn by hearing other people speak correctly and trying to predict which morphemes will occur next, then the structure of the information they receive parallels that provided by our task (see Elman, 1990). There is clearly more going on in natural language processing than next morpheme prediction—for example, speakers may be mapping from form to meaning, or deciding on courses of action. We suggest, in keeping with many previous connectionist studies, that prediction is one element of what people learn.

3. Simulation

3.1. Architecture and dynamics

The network had two layers, a feature layer (the input layer) and a class layer (the output layer; Fig. 2). The feature layer was feedforward connected to the class layer and the class layer was recurrently connected. Each of the five classes was assigned an indexical bit representation in the class space (one unit set to 1, the rest 0). We call such a network a two-layer classification (2LC) network. The activation dynamics for the (recurrent) class layer are specified in Eqs. 1a and b. Here, a_i is the activation of the i'th unit, w_{ij} is the weight from the j'th unit to the i'th unit, and τ_a is a time constant of the activation dynamics.

$$net_i = \sum_{j \in Class\ Units} a_j w_{ij} + \sum_{k \in Input\ Units} a_k w_{ik} \tag{1a}$$

$$\tau_a \frac{da_i}{dt} = net_i \cdot a_i \cdot (1 - a_i) \tag{1b}$$

Learning was accomplished by a simple version of the delta rule:

$$\tau_w \frac{dw_{ij}}{dt} = (t_{ip} - a_{ip})a_{jp} \tag{2}$$

Here, t_{ip} is the target for the i'th Class unit when the p'th pattern has been presented on the input layer and τ_w is the learning time constant or inverse learning rate. We used this very simple approach because it worked well with very little tuning of parameters and avoids the computationally expensive error signal propagation associated with other recurrent network training methods (see next section).

3.2. Training procedure

On each of 16,000 trials, a pattern was selected at random from the 20-pattern training set. We chose 16,000 because the network's error rate was generally stable by that time. The input units were clamped to the feature values specified by the pattern. The output units were set to a vector with all equal activations (unbiased among the five classes; the value of each activation was set to 0.1). The time evolution of the continuous equations was approximated by Euler integration with a step size of 1, the current standard practice in connectionist language modeling.[4] We set τ_a = 10 and τ_w = 100. While the input was clamped, the class units settled for 10 time steps. Then the inputs were all set to zero and the class units settled for an additional 40 time steps.

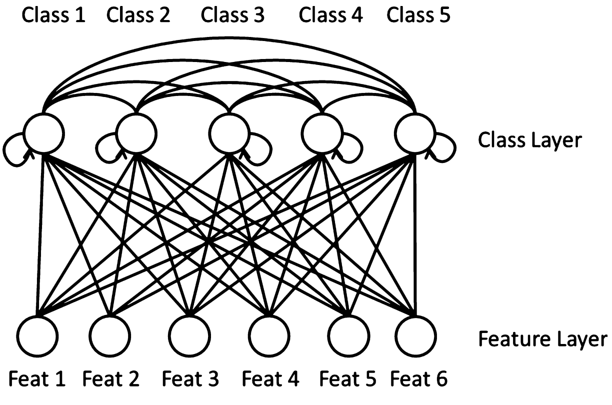


Fig. 2. Architecture of the network. The feature layer is the input layer. It feeds forward to the recurrently connected class layer, which is the output layer.

The network was started, at the beginning of training, with all weights equal to zero. In typical recurrent connectionist simulations, changes in the activations and the weights alternate usually with many steps of activation change, then one step of weight change. In this simulation, the weights and activations were changed simultaneously. As the time constant for weight change is larger than the time constant for activation change, this simultaneous update mechanism has fast activation change and slow weight change, as in the alternating method. The method has the property that the weights grow without bound, although the error and the relationships between them appear to stabilize in successful learning runs.

We updated the weights and activations simultaneously for two reasons: It is mathematically simple and it worked well. When we ran the network with alternating activation and weight change, it failed to correctly classify the 20 input patterns even after extensive training. This seems to be because the small changes to the weights near the beginning of the training process create very complex, small magnitude dynamics. The network moves in an erratic fashion, and by the end of each trial, it has lost most of the information that was provided by the input. An alternative approach, which would also make sense, is to compute the dependency of the error at times selected for update on the weights across time and to use this information to adjust the parameters in the optimal direction as, for example, in backpropagation through time (Rumelhart, Hinton, & Williams, 1986), continuous backpropagation through time (Pearlmutter, 1995), real-time recurrent learning (Williams & Zipser, 1989), or recurrent backpropagation (Pineda, 1995). Our method may be thought of as a simple method of approximating the integral of dependency over time by computing an error and adjusting at each time point. Another alternative is to use the state of the system at a well-chosen time point prior to the arrival of feedback to compute weight changes (Oppenheim, Dell, & Schwartz, 2010; Tabor, Juliano, & Tanenhaus, 1997).

Another distinctive and important feature of our training method is that we continue training (for 40 time steps) after the inputs have been turned off. We did this to encourage the network to autonomously stabilize on the correct choice, once the input propelled it in the direction of that choice. This feature corresponds to the assumption that humans continue to strengthen their tendency to produce the correct action, even after the input has disappeared—a choice that seems reasonable in light of the many studies in which humans respond sensibly to stimuli after they have disappeared from the environment (e.g., Sperling, 1960). This choice allows us to discover the organizing principles of the network by studying a single autonomous dynamical system, making it feasible to use simple, standard methods of analysis. If one preferred to keep the input continuously on, then one could employ open dynamical systems approaches to analysis (Hotton & Yoshimi, 2011). For our stimuli, keeping the input on throughout each trial produced a similar pattern of results with respect to staging and generalization, except that the extent of a critical generalization tendency (non-similarity–based generalization of the default class—discussed in Section 3.4.3 below) was much reduced.

We ran the network 30 times, starting each time with all weights zero. The runs had different profiles because of random selection of the ordering of the trials. We chose the number 30 to match the number of participants who achieved effective learning in the human study (see Section 4.2.2), based on the notion that if the model variability turned out to be much greater than the human variability, then we might question whether it is a good model. In fact, the evidence points in the other direction: The model variability is much lower than the human variability. We thus think of the model sample as approximating ideal performance much more closely than the human sample does.

3.3. Predictions and model assessment

We were interested in the staging and generalization behaviors of the network model. Following the work on learning the English past tense, we thought the model would show an initial period of correct behavior on a subset of strong and default items, followed by a stage of overregularization (strong items assimilate to default), followed by a stage of all correct behavior. We planned to examine generalization by considering bit vectors in the input space that were not part of the training set. Again following results for natural language cases, we expected generalization to exhibit an asymmetry: We expected novel items to be assigned to strong classes only if they were similar to the strong classes; in contrast, we expected the default class to operate according to an Elsewhere Principle, with novel items assigned to the default class whether they were similar to it or not. To define the similarity, s, between two input vectors, $\boldsymbol{x}$ and $\boldsymbol{y}$, we used a metric due to Shepard (1987):

$$s(\vec{x}, \vec{y}) = e^{-|\vec{x}-\vec{y}|^2}, \tag{3}$$

where |…| denotes the Euclidean norm.

In our case, similarity is a function of the distance in the feature space, consistent with the intuitive notion of similarity invoked in discussion of morphological marking (e.g., two stem forms like “sing” and “bring” are considered similar in virtue of their shared phonological features). We define the similarity between an item and a class to be the sum of the similarities between the item and the class members. This definition is aligned with the generalized context model (GCM; Nosofsky, 1986) where the probability of assigning an item to a class is proportional to the sum of the similarities between the item and the class members.

3.4. Results

We first describe how well the models learned the training data, and then report staging and generalization behaviors.

3.4.1. Training effectiveness

To test each model’s performance on the training set, we fixed its weights to the values at the end of the 16,000 training trials and presented each training vector, letting the activations settle as in the training process. To assess classification accuracy, we considered the final state of the class layer after a test trial. We normalized the final activation vector of each trial by dividing each class activation by the sum of the class activations. Accuracy was taken to be the dot product of this normalized vector with the correct indexical bit vector. This is tantamount to interpreting the end state as a probability distribution and taking the accuracy to be the expected rate at which the correct bit would be chosen from this distribution. In 18 of the 30 runs, the trained model had over 99.9% accuracy on every training pattern. In 6 of the 12 remaining runs, it incorrectly classified one of the four default patterns (putting it in a strong class). In the other six erroneous runs, it activated both the default unit and a strong-class unit for one of the default exemplars. These results indicate that the models learned the training data well, and when they made errors, the errors were systematic.

3.4.2. Staging

To examine staging, we considered the distribution of classifications over the course of training. Fig. 3 shows the results for one network (the other networks behaved very similarly). The model exhibits three phases of learning: a brief early phase when its behavior is at chance on all trials, a middle phase when the strong classes are correct but the default class is in error, and a final phase when all classes are correctly classified. This result contrasts with our prediction based on the English past tense: There is no evidence of an early phase of correct behavior on a subset of strong and default items (classifications seem to be selected at random in the earliest phase of the model's behavior); moreover, the strong-class items achieve reliably correct behavior *before* the default-class items. One might take this result as an indication that the network is a poor model of human processing; however, we believe it stems from the structure of the training data on which the model was trained. Thus, we predicted that humans, trained on the same data as the model, would show a similar staging pattern (see Section 4).

We hypothesized that, during the phase when the default-class members were inaccurately classified, they were being put into strong classes on the basis of similarity (in keeping with natural language findings that strong [irregular] classification is similarity based). To explore how the model was behaving when it was not accurate, we recorded, for each input, the normalized output activations, which we interpret as probabilities of class assignment. Fig. 4 shows the evolution of these probabilities over training for each default pattern. The figure indicates that the default class was first randomly classified, then assimilated to the strong classes ("overirregularization"), and finally switched to correct classification. The figure also indicates that, during the middle, assimilation phase, in keeping with our predictions, a misclassified default pattern was generally assigned to a maximally similar strong class (similarity defined as in Section 3.3). The strong classes, on the other hand, were misclassified as defaults only infrequently during the very brief random classification period at the beginning of training.

3.4.3. Generalization

To examine generalization in the model, we tested our 30 test networks' responses to all the non-trained bit vectors in the input space. The networks were trained on 20 bit vectors

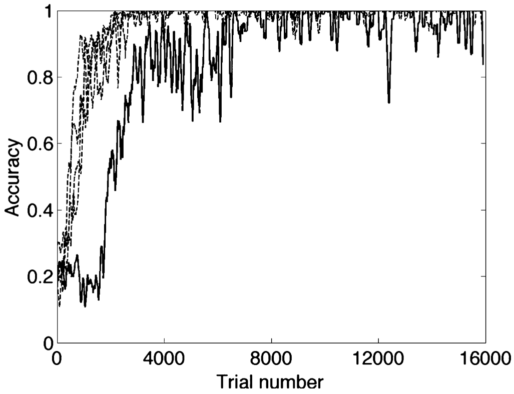


Fig. 3. Classification accuracy for the model versus time. Accuracy is the value of the correct bit in the normalized class layer activations. Each curve shows the average of four exemplars, which belong to one training class. Dotted lines: strong classes. Solid line: default class.

Default training pattern	Closest strong class
17	1 and 2
18	1 and 3
19	1 and 2
20	4

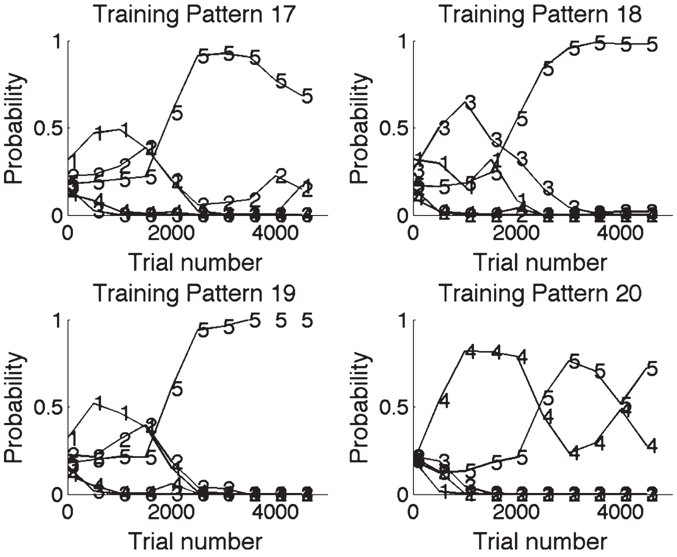


Fig. 4. Assimilation behaviors of individual default-class members. The curves labeled 1–4 indicate assignment to the strong classes. Curve 5 indicates default assignment, the correct response. The most similar strong class or classes for each default exemplar are shown in the table. For clarity, the figure only displays the first 5,000 time steps of training.

Table 3
Network generalization behavior

Network Results	Class 1	Class 2	Class 3	Class 4	Class 5 (default)	Total
1. All 44 novel test items: counts	61	37	300	64	858	1320
2. NSBG all 44: counts	0	0	0	0	738	738
3. Proportions (odds) of NSBG	0 (0)	0 (0)	0 (0)	0 (0)	0.86 (6.14)	0.56 (1.27)
4. The 16 novel items given to humans: counts	3	8	129	14	326	480
5. NSBG 16: counts	0	0	0	0	260	260
6. Proportions (odds) of NSBG 32	0 (0)	0 (0)	0 (0)	0 (0)	0.80 (4.00)	0.54 (1.17)

Notes. NSBG, Non-similarity–based generalization. Note that SBG (similarity-based generalization) = count – NSBG in each column. The proportions of NSBG, shown in Rows 3 and 6, give p = NSBG/(NSBG + SBG) for each class. If p is the proportion of NSBG for a class, then the odds of NSBG for that class are $p/(1 - p)$. Rows 4–6 anticipate a parallel analysis of data collected from human participants. These data are shown in Table 6 below.

in the six-dimensional input space, so there were 44 novel cases. Among these 44, 37.2 (85%) on average produced outputs that were closest (Euclidean distance) to one of the indexical bit vectors. The remaining inputs produced outputs closer to one of the other unit hypercube corners,[5] although no trial converged to a state closest to the origin. The first row of Table 3 shows the total number of novel inputs assigned to each class over the 30 runs (on each trial, we normalized the activations at the end of the trial to specify a probability distribution and then drew one random sample from this probability distribution to select a class). The model showed a strong preference for the default class, although it also often picked Class 3.[6]

As noted above, a system that employs similarity in combination with an Elsewhere Condition will place novel items in strong classes on the basis of similarity, but it can place novel items into the default class whether or not they are similar to exemplars of the default class (Bybee & Moder, 1983). For the sample of classifications tallied in the first row of Table 3, we asked how many instances were assigned to a class different from the one(s) it was most similar to. Such instances of non-similarity–based generalization ("NSBG") are tabulated in row 2 of Table 3 (row 3 shows proportions and odds). Indeed, the distribution of the network's classification is more consistent with an Elsewhere Condition mechanism than with a pure similarity-based mechanism: Both the strong and default classes show similarity-based generalization ("SBG"), but only the default class shows NSBG. Rows 4 through 6 of Table 3 focus on a subset of novel items on which we tested human participants (see Section 4 below).

In sum, the model shows a qualitative difference between the strong classes and the default class in staging (delayed default stabilization) and generalization (non-similarity–based generalization restricted to the default class). In the next section, we ask whether humans show a similar pattern.

4. Human experiment

4.1. Method

4.1.1. Participants

Eighty-eight undergraduate students at the University of Connecticut participated in the experiment for course credit. One participant's data were excluded from the analysis because the program failed to log some needed information.

4.1.2. Materials

We used the same 20 six-element vectors as in the simulation study to generate 20 novel creatures (see Section 2 above). To avoid a systematic difference between strong-class and default-class creatures in terms of visual salience, the mapping between the vector elements and the six manipulated body features (antenna, legs, mouth, nose, tail, and wings) was randomized across participants.

The 20 creatures were repeatedly presented to participants during the training phase (see 4.1.3 Procedure). In the subsequent test phase, we presented eight trained items (one exemplar from each strong class, along with the four default exemplars[7]). We also created eight additional creatures (Table 4) to probe generalization in the test phase. The two additional (non-core) features were assigned randomly to each of these patterns for each participant. The eight training items were included in the test phase to assess the stability of learning. The eight novel items were included in the test phase to assess generalization behavior. We presented each of the 16 test items twice (see 4.1.3 below) to improve reliability of the test. We did not include more than these 32 test trials because we wanted to avoid forgetting and inattention to the task. As it happens, the task proved quite difficult for many participants.

Table 4
The features of the eight additional creatures presented during the testing phase

Pattern No.	Features
Test 1	0000RR
Test 2	0001RR
Test 3	0110RR
Test 4	0111RR
Test 5	1001RR
Test 6	1011RR
Test 7	1101RR
Test 8	1111RR

Note. "RR" stands for two random bits.

4.1.3. Procedure

The experiment consisted of two phases. In the first (training) phase, participants learned the classifications of 20 creatures. Each trial started with a 500 ms presentation of a fixation cross. The cross was then replaced with one of the 20 creatures. Participants classified the creature into one of five classes by pressing a number key among 1–5; the mapping between the number keys and the classes was randomized across the participants. When the participant pressed a correct key, the statement "Correct !!" appeared in blue above the creature for 1 s. When the participant pressed an incorrect key, the statement "The Correct Answer is *N*" (where *N* was the number of the correct class) appeared in red above the creature for 2 s. At the end of every block of 20 trials, participants were shown their block mean accuracy and were offered the chance to take a short break before continuing the task. In each block, each of the 20 training creatures was presented once and the presentation order was randomized. The study phase ended when the block mean accuracy was equal to or greater than 0.9 for three blocks in a row, or when 25 blocks were completed.

After finishing the study phase, a new set of instructions appeared, explaining that the participants would view some additional creatures and would be asked to classify them, but no feedback would be given. The instructions also explained that some of the creatures to be presented in this phase did not appear in the first phase and that one should classify them according to one's first impulse. In the test phase, the 16 test creatures were presented twice, once in each of two blocks, with different, random ordering of the test items in each block. The whole experiment took about 50 min.

4.2. Results

Among 87 participants, 31 satisfied the accuracy criterion for successful learning during the training phase (i.e., their block mean accuracies were not less than 0.9 in the last three blocks).[8] As our model is concerned with the staging and generalization under the assumption of successful learning, we focused the response patterns of those 31 participants.

4.2.1. Staging

Figure 5 shows the learning trajectories of two sample participants in the training phase. We modeled the change in classification accuracy (statistically) as a function of BlockNumber and ClassType (Strong vs. Default). To handle the binary accuracy data clustered within each individual, we used the generalized linear mixed model with a logit link function (Jaeger, 2008). First, we constructed a base model (Model 1) where the log odds of classification accuracy were modeled as a linear function of BlockNumber. BlockNumber was adjusted to make the first block have the value zero. To test the effect of ClassType on the accuracy change, we then constructed two more models by adding a new predictor ClassType to the base model (Model 2) and adding an interaction term ClassType-by-BlockNumber to Model 2 (Model 3). The model comparisons revealed that (a) the intercept varied with ClassType, $\chi^2(1) = 8.783$, $p = .0031$; and (b) the slope (i.e., the coefficient of BlockNumber) also varied with ClassType, $\chi^2(1) = 7.262$, $p = .0071$. The odds of correct classification of default-class exemplars were 0.129 (= $e^{-2.047}$) at the first block and increased by a factor of 1.228(= $e^{0.205}$) (23%), on average, in each block. For strong-class exemplars, the odds of correct classification was 0.518 (= $e^{-2.047 + 1.388}$) and increased by a factor of 1.300 (= $e^{0.205 + 0.057}$) (30%) in each block. Both the intercept difference and the slope difference were significant. The parameter estimates from the final statistical model (Model 3) are shown in Table 5. The results suggest that participants learned the default-class exemplars later and more slowly than the strong-class exemplars.

Based on these results, we claim that the human learning, like the network learning, can be naturally divided into three stages: In the first stage, learners have no knowledge of classes. Thus, classification accuracy is low for both strong and default classes. In the second stage, learners correctly classify strong-class creatures but

Table 5
Summary of the fixed effects in the mixed logit model ($N = 11{,}240$; log likelihood = −5,385): human training phase

Predictor	Coefficient	*SE*	Wald *Z*	*p*
Intercept	−2.0468	0.3057	−6.695	<.0001
BlockNumber	0.2054	0.0240	8.579	<.0001
ClassType = *strong*	1.3884	0.3169	4.382	<.0001
Interaction = *BlockNumber & ClassType*	0.0567	0.0195	2.915	.0036

Note. The analysis provides evidence that the default class is learned later and more slowly than the strong classes.

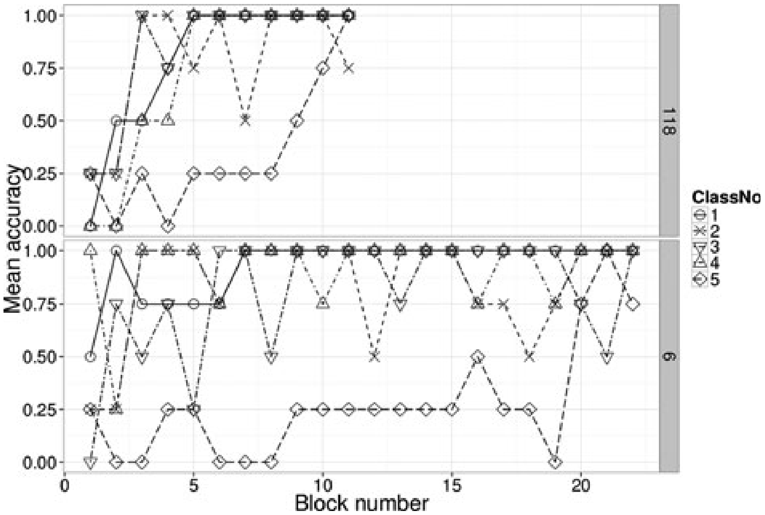


Fig. 5. Accuracy versus time for two human participants. The rates of correct assignment of training stimuli to classes across blocks for Participants 118 and 6. Classes 1–4 are the strong classes. Class 5 is the default class.

misclassify default-class creatures as strong-class items (default-to-strong assimilation or “overirregularization”). In the third stage, learners correctly classify both strong- and default-class creatures.

As the network exhibited similarity-based default-to-strong assimilation during the middle stage (see 3.2 above), we hypothesized that the humans would as well. To test this hypothesis, we analyzed participants’ responses to four default-class creatures over the blocks where the block mean accuracy across all 20 creatures was between 0.5 and 0.9 (separately assessed for each participant). This period, we assumed, corresponded to the second stage. During this period, the mean default-class accuracy was 0.406 (*SD* = 0.158) and the mean strong-class accuracy was 0.814 (*SD* = 0.081).

As the human data were sparser and noisier than the network data, we used a statistical technique to assess the presence of similarity-based default-to-strong assimilation. The responses to the four default-class creatures were classified into three categories: (a) correct classification into the default class; (b) incorrect classification into the most similar (strong) class(es); and (c) incorrect classification into less similar classes. Similarity was determined as in the simulation study (Table in Fig. 4). For each individual, we computed the proportion of misclassifications into the most similar strong class(es) (P_S), the proportion of misclassification into the less similar strong class(es) (P_{NS}), and the logarithm of the ratio of the proportions, log(P_S / P_{NS}). If participants choose the most similar strong classes more than less similar strong classes when they misclassify default-class creatures, log(P_S / P_{NS}) will be greater than zero. The measure was not defined for three participants because, for these participants, P_{NS} was zero. In this case, we substituted the other participants’ maximum value of log(P_S / P_{NS}). With respect to our test of the similarity-based assimilation hypothesis, the substitution of the other participants’ maximum for cases in which P_{NS} is zero is conservative because it underestimates the observed rate of similarity-based assimilation. A *t* test showed that the mean value of the measure was significantly greater than zero, $t(30) = 6.38$, $p < .0001$ ($M = 1.145$, 95% CI = [0.778, 1.511]), in keeping with the similarity-based default-to-strong assimilation hypothesis.

In sum, 31 participants who learned 20 creatures well in the first phase seemed, like the networks, to go through three distinguishable stages, including similarity-based assimilation of default strong-class creatures to strong classes in the second stage.

4.2.2. Generalization results

Recall that, in the test phase, we presented eight old creatures and eight novel creatures twice to test stability of learning generalization behavior.

To assess stability of learning, we inspected the accuracies on old items. One participant was excluded from the generalization analysis because that participant’s mean accuracy for old items was very low (0.375) and distant from the other participants’ mean accuracies. For the remaining 30 participants, the mean of the individual mean accuracies for the eight old items was 0.810 and the standard deviation was 0.095. A *t*-test revealed that the mean accuracy for the default class (0.675) was significantly less than the mean accuracy for the strong classes (0.946), $t(29) = 5.879$, $p < .0001$, $M_{accS-accD} = 0.271$.

This result suggests that even though the overall accuracy was above 90% at the end of training, the default-class encodings were relatively unstable.

Generalization behavior for the 30 participants was inspected as in the simulation section. Based on the network, we hypothesized that when a novel item is classified into a strong class, the classification should be similarity based, whereas the generalization into the default class can be non-similarity based. To test this hypothesis, we dichotomized participants' classification of novel items into (SBG) and (NSBG). Table 6 summarizes the results (compare Table 3). The first row gives overall counts of classification of novel items into different classes, collapsed across participants. Like the networks, the participants as a group showed a preference for Class 5 and Class 3 over the other classes. Row 2 tabulates (NSBG). Row 3 shows proportions and odds. Participants showed a much higher rate of NSBG for the default class than for the strong classes. Model comparison revealed that the log odds of NSBG were a function of ClassType, $\chi^2(1) = 12.85$, $p = .0003$.

The odds of NSBG when novel test items were classified into the default class were 4.482 (= $e^{1.500}$) (i.e., the proportion of NSBG was 4.48 times higher than the proportion of SBG). The parameter estimates from the preferred statistical model of the human data are shown in Table 7.

The odds of NSBG for strong classes were 0.450 (= $e^{1.500-2.299}$), approximately a 10-fold difference in the direction predicted by the network model.

Table 6
Human generalization behavior (compare to Table 3, which shows corresponding model data)

Human Results	Class 1	Class 2	Class 3	Class 4	Class 5 (default)	Total
16 novel test items: counts	47	51	173	61	148	480
NSBG: counts	21	15	53	14	121	224
Proportions (odds) of NSBG	0.45 (0.81)	0.29 (0.42)	0.31 (0.44)	0.23 (0.30)	0.82 (4.48)	0.47 (0.88)

Note. NSBG, Non-similarity–based generalization. Note that SBG (similarity-based generalization) = count − NSBG in each column. The proportions of NSBG, shown in Rows 3 and 6, give p = NSBG/(NSBG + SBG) for each class. If p is the proportion of NSBG for a class, then the odds of NSBG for that class are $p/(1 - p)$. (Compare to Table 3.)

Table 7
Summary of the fixed effects in the mixed logit model ($N = 480$; log likelihood = −275.9): human test phase

Predictor	Coefficient	*SE*	Wald *Z*	*p*
Intercept	1.5000	0.2128	7.047	<.0001
ClassType = *strong*	−2.2989	0.2437	−9.435	<.0001

Note. The analysis provides evidence that greater NSBG occurred in the default class than in the strong classes.

4.3. Discussion

The humans, like the networks, exhibited a three-stage learning pattern and a generalization pattern consistent with the Elsewhere Condition, and inconsistent with pure similarity-based classification. In a real language development context, stages could be caused by genetically triggered maturational changes that make new mechanisms available. Such an explanation is not plausible in an artificial category learning context, so we would like to discover another explanation of the stages. In children's learning of the English past tense, three stages have also been documented, but the first two stages are distinct from those observed here: The first English past tense stage involves accurate performance on a small subset of items, whereas the current first stage involves random performance on all items; the second past tense stage involves strong assimilation to the default classes ("overregularization"), whereas the current second stage involves default assimilation to the strong classes ("overirregularization"). We would like a theory that predicts these different staging patterns as a function of the differences between the two developmental situations.

Regarding generalization behavior, the bias toward non-similarity–based classification into the default class suggests that an elsewhere principle may have been at work in the humans. If the humans learned how to classify the members of the default class simply by memorizing each default exemplar, then they would not show a reliable tendency to assign default status to novel items that are more similar in their features to strong-class items. Thus, the outcome we observed does not simply follow from the fact that the participants learned the training data. It is possible that a biologically specified mechanism makes the elsewhere mechanism available for use in this category learning task as well as in natural language systems. However, if the elsewhere behavior could be derived from principles needed independently to explain how the participants learn the categories in the first place, the explanation would be more parsimonious.

In the next section, we employ tools of dynamical systems theory to understand how the staging and the generalization behavior come about in the networks, which have no maturational or elsewhere mechanisms except to the extent that these are embodied by weight change and weight values.

5. Analysis of the model

5.1. Introduction to analysis tools

Three technical concepts form the foundation of our analyses: (a) equilibria and stability; (b) control parameters and bifurcations; and (c) the Voronoi partition. We introduce them here.

5.1.1. Equilibria

A helpful step in relating a dynamical system in a continuous space to a symbolic model can be to carve up the space into subsets that are related to the operation of particular rules. Identifying equilibria helps with this goal. A state, **a**, of a continuous-time dynamical system is called an *equilibrium* if, when the system is at **a**, it stays at **a** forever. The equilibrium **a** is *asymptotically stable* if, when the system is placed near **a**, it converges to **a** (in this case, **a** is called an *attractor*). It is *unstable* if there are points arbitrarily close to **a**, from which the system leaves any sufficiently small neighborhood of **a**. As we noted in Section 1.3, the set of states from which the system converges to an attractor **a** is called the *basin of attraction* of **a**. We expected the (asymptotically) stable equilibria to be associated with rules (in the sense of Kiparsky, 1973 and Albright & Hayes, 2003). Recall that on each trial the model is stimulated by an input for 10 time steps and then the input is turned off. The system is then supposed to converge to an indexical bit vector. Because attractors are associated with convergence of a dynamical system's state, it is natural to hypothesize that the indexical bit vectors of the successful, trained network are attractors (i.e., asymptotically stable equilibria) of the class layer dynamics. (We can ignore the input layer because, by the end of a trial, it has been turned off.) We investigate this hypothesis below.

The equilibria of the class layer dynamics are those points for which

$$\frac{da_i}{dt} = \frac{1}{\tau_a} net_i \cdot a_i \cdot (1 - a_i) = 0 \qquad i \in \{1, 2, 3, 4, 5\} \tag{4}$$

(i.e., points at which the activations are not changing). There may exist many solutions to this set of equations. It is easy to see that all the corners of the class space unit hypercube, including the indexical bit vectors, are always among them.

We have hypothesized that stable indexical equilibria correspond to rules. In the present system, these equilibria are always present, but they are not always stable. Thus, to examine the emergence of the network's "rule" system, we must track the stability of the indexical equilibria over time. To determine the stability of the equilibria, we examine the eigenvalues of the Jacobian of the class layer activation dynamics evaluated at each equilibrium.[9] If the real parts of the eigenvalues associated with an equilibrium are all negative, then the equilibrium is asymptotically stable (i.e., it is an attractor). If at least one eigenvalue has a positive real part, then the equilibrium is unstable (it is not an attractor). If an equilibrium has some eigenvalues with zero real parts, then it is called a *non-hyperbolic equilibrium* and its stability characteristics cannot be determined from the eigenvalues alone. Thus, a sufficient condition for stability is that the eigenvalue with maximal real part has a negative real part. For this reason, we tracked the maximal real parts of the eigenvalues over time. We are interested here in general stability (not just hyperbolic stability), but in our model, non-hyperbolicity was rare: It occurred for an instant at the beginning of training when all weights were zero, and at a few instants of time during training—we tracked it and we make note of where it occurs below, but we are able to draw our main conclusions based on analysis of the hyperbolic equilibria.

5.1.2. Control parameters and bifurcations

The activation dynamics of the 2LC ("Two Layer Classification") network can be described at two levels: the level of the state variables and the topological level. At the state variable level, there is a vector of activations whose values change continuously over time. It is common to define a metric on the state space, so distances between states and rates of change can be evaluated. The topological description, on the other hand, characterizes the asymptotic behavior of the system in the limit of infinite time; it identifies the number and stability characteristics of equilibria; it describes the configuration of attractor basins and the shapes of their boundaries. It is via the topological description that dynamical systems are most naturally related to symbolic models of cognition. We explore this relationship below.

Parameters that specify the state dynamics are called *control parameters*. In the present case, the weights are control parameters. Sometimes, continuous change in control parameters produces qualitative change at the topological level. For example, an equilibrium might shift from being unstable to being stable, or the number of equilibria might change instantaneously. Such qualitative shifts associated with continuous control parameter change are called *bifurcations*. The learning rule induces continuous change in the control parameters. Building on our hypothesis above that by the end of training the indexical bit vectors have become stable equilibria, we further hypothesize that they undergo bifurcations during the course of learning, switching from unstable to stable.

5.1.3. The Voronoi partition

Given a discrete set of points (called "anchor points") in a metric space,[10] the Voronoi partition divides the space up into a set of regions, where each point in the space is in one region or on the boundary between one or more regions and all the points in a particular region are nearer to that region's anchor point than to any other anchor point. That is, given a metric space X with metric d and a countable set of points in it, $A = \{a_1, a_2, a_3, \ldots\}$, the *Voronoi partition* of X induced by A is the set of subsets $A_1, A_2, A_3, \ldots$ of X satisfying $x \in A_i \Rightarrow d(x, a_i) < d(x, a_j)$ for $j \neq i$. The members of A are called the *Voronoi anchor points*. The *boundary of the partition* is the set of points in X that are not in A_i for any i. The Voronoi partition and its boundary are not a standard construct of dynamical systems theory, but they are relevant in the present context. We suggest that near the initial state, the Voronoi boundary induced by the strong-class indexical bit vectors approximates the complement of the union of the strong classes, the set that Hare et al. (1995) proposed as a manifestation of the Elsewhere Condition in their model.

5.2. Analysis of the model's staging behavior

Recall that if the real parts of all eigenvalues associated with an equilibrium are negative, then the equilibrium is asymptotically stable. Fig. 6 is a graph of the maximal real parts of the eigenvalues of the indexical bit vectors, with circles around all points where the corresponding equilibrium was non-hyperbolic. The figure shows that all the indexical bit vectors become asymptotically stable over the course of training. The stabilization

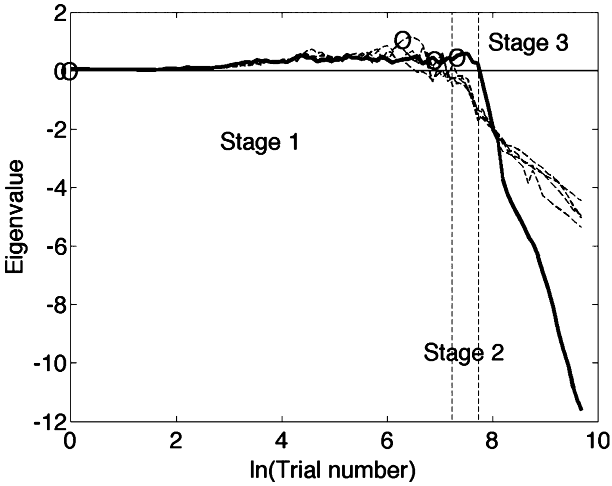


Fig. 6. Maximal real parts of eigenvalues versus time for the indexical bit vectors of the class units. Circles indicate cases where some real parts were very close to zero (within 0.0001). The dotted curves correspond to the strong classes; the solid curve corresponds to the default class. The stage boundaries (indicated by vertical dotted lines) are determined from these data (see text). Note that time (trial number) is shown on a log scale.

occurs in three stages: First, no indexical bit equilibria are stable. Then, the strong classes stabilize at roughly the same time. Later, the default class stabilizes.

This analysis confirms our hypothesis in Section 5.1 that the indexical bit vectors undergo bifurcations as the weight control parameters change during the course of learning, becoming stable by the end of training. The same pattern occurred in every run we tested.[11] Moreover, the eigenvalue results line up with the staging results for the model reported in Section 3: The stabilization of the strong classes coincides with the rise of strong-class accuracy and the onset of default assimilation to strong classes; the stabilization of the default class coincides with the switch to correct classification of default exemplars. Based on this analysis, we use the eigenvalue zero crossings of the maximal eigenvalue to associate time points with the stage transitions in the discussion henceforth. We take the Stages 1–2 transition to occur at the mean final zero-crossing time for the strong classes, and we take the Stages 2–3 transition to occur at the final zero-crossing time for the default class.

A related development occurs in the Input (i.e., Feature Layer) → Class weights. Recall that the class unit activations start off small (near the origin) at the beginning of a trial. Therefore, at this time, the second term in Eq. 1a dominates. When the dot product of the Input → Class weights with the input vector for one class unit, *i*, is larger than that of any other class unit, then, with the input turned on, the class vector is initially driven approximately toward the indexical bit vector of the *i*'th class. Fig. 7 shows the degree to which the input drives the class activations toward the appropriate indexical bit vector over the course of training. This figure coincides partly with the staging results reported in Section 3: The strong exemplars achieve positive values by the middle of the second stage,

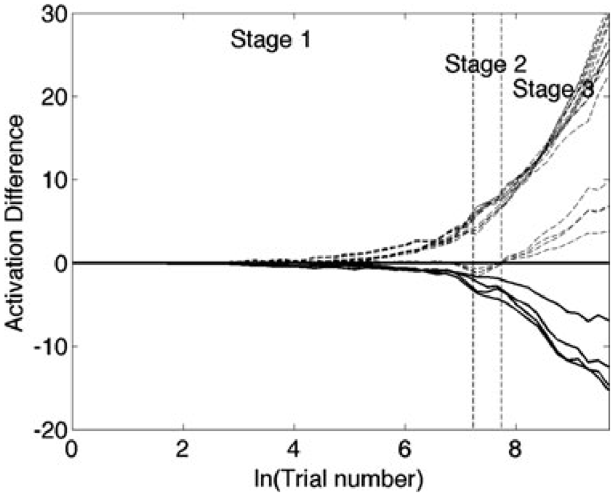


Fig. 7. For each class unit on each training trial, we computed the product of the Input → Class weights for the class unit and the training pattern (second term in Eq. 1a). Taken as a vector, these five values constitute a driving signal provided by the training pattern. We then computed the difference between the element of this vector corresponding to the correct class and the highest value among the remaining elements. A high positive difference indicates that the input is driving the network toward the correct indexical bit vector. A negative difference indicates that the input is driving the network toward an incorrect indexical bit vector or a non-indexical corner. The graph plots these differences versus time. Dashed lines correspond to strong-class exemplars, and solid lines correspond to default-class exemplars.

but the default exemplars never do. This lack of helpful input propulsion for the defaults helps drive the late extensive growth of the default attractor basin discussed below.

The combination of the eigenvalue changes and the Input → Class weight changes provides a fairly complete picture of how the network generates the observed staging behaviors. During the first phase, every indexical bit vector has an eigenvalue with positive real part, so all the indexical bit vectors are unstable. During this phase, the Input → Class weights are small and do not drive the network close to any bit vector, so behavior is random. During the second stage, the fixed points corresponding to the strong classes have all stabilized, and for strong classes, the input drives the class activations toward the correct indexical bit vector. When the input is turned off, the input is in the attractor basin of the correct class and the recurrent dynamics cause it to approach the correct vector. During this stage, the default indexical bit vector is unstable, so it is very unlikely that the system will converge on it; instead, the strong indexical bit vectors capture all trajectories, including those of default exemplars. Thus, we observe default assimilation to strong classes (overirregularization). Finally, at the start of the third phase, the indexical bit vector corresponding to the default class stabilizes and its basin becomes shaped so that it includes the endpoint of the input driving process for each default-class member.

5.3. Analysis of the model's generalization behavior

Figure 7 above indicates that although the directional guidance coming from the input steers the class layer activations in the right direction for the strong classes, the default input is not directly helpful. Nevertheless, *some* property of the default inputs must allow the network to recognize them as it correctly classifies them. The Voronoi partition, introduced in Section 5.1, is relevant here. We are interested in the path of the network from its initial state near the origin toward the indexical bit corners of class space. We define the Voronoi partition on the basis of Euclidean distance in the class space, taking the Voronoi anchor points to be the strong-class indexical bit vectors. We hypothesize that the network solves the default exemplar identification problem by letting the default inputs drive the class units along the boundary of this Voronoi partition while the strong inputs drive the network away from it. (Recall that, on each trial, the network starts at a point near the origin [viz., 0.1, 0.1, 0.1, 0.1, and 0.1]; this lies on the Voronoi boundary.)

To test this hypothesis, we considered, for each input pattern, *inputp*, the input driver $idp = [\sum_{k \in Input\ Units} a_k w_{ik}]$ (see Eq. 1a). Here, the notation [] indicates the vector of values formed by taking $i = \{1, 2, 3, 4, 5\}$. We then determined, when the system first moves, under the influence of this driver, away from its initial state in class space, which two anchor points the system became closest to, and computed the vector difference, *vdiff*, of these two anchor points. Next, we computed the cosine of the angle between *idp* and *vdiff*. The absolute value of the cosine indicates how aligned the system is with the proximal portion of the Voronoi boundary.

Figure 8 shows the absolute value of cosine versus time for each input pattern for a sample network (every run was similar). The transition from Stages 1–2 is associated with a separation of some of the strong drivers from the default drivers, with the default drivers closer to the Voronoi boundary. The transition from Stages 2–3 completes the separation, again with the default drivers closer to the boundary. This result is consistent with our hypothesis that the input drives the default exemplars along the boundary. It is important to note that the default items could, in principle, be driven directly toward the default equilibrium, once it has stabilized. In that case, if we include the default indexical bit vector as a Voronoi anchor point, then the drivers of such default items should drive the system away from the modified boundary. However, using this alternative boundary produces no difference in the Voronoi profile. This result is consistent with our claim that the default class is recognized by an elsewhere mechanism, not by detecting specific properties of the default inputs.

How does the default-class attractor capture the trajectories that are near the Voronoi boundary? Fig. 9 shows that the Stages 2–3 transition is marked by strong growth in the default self-weight from a value near zero to a value close to the mean of the strong-class self-weights. During the third stage, the default self-weight magnitude rises above all the strong self-weight magnitudes. The thin solid curves in the figure indicate that the non–self-connections are negative during most of the training process. We hypothesized that, as the self-weight grows, the attractor basin of the default indexical bit vector comes to

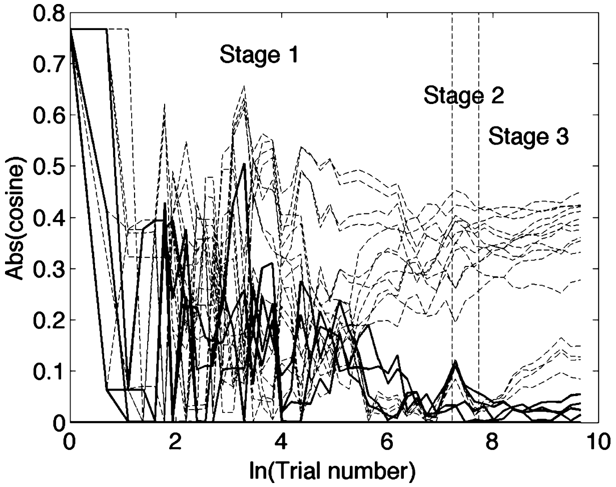


Fig. 8. Divergence of the input driver from the Voronoi boundary in the class space at t_0 for each trial of training (0 = alignment with the boundary, 1 = maximal divergence). The Voronoi anchor points were the strong indexical bit vectors. Dotted lines correspond to strong exemplars; solid lines to default exemplars.

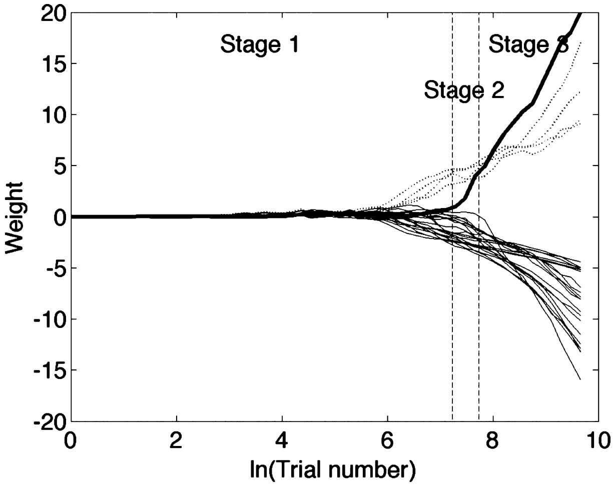


Fig. 9. Class weights over the course of training. The thick solid curve shows the default self-weight. The dotted curves show the strong-class self-weights. The remaining curves show the interweights (w_{ij}, for $i \neq j$).

encompass more and more of the input space under the network map. To test this hypothesis, we set the network weights to the values they had at the Stages 2–3 transition, and then increased the default-class self-weight while keeping all other weights constant. This is an example of a control parameter manipulation, defined in Section 5.1. In this case, the default-class attractor progressively captured every input corner, providing evidence that its pre-image (the set of inputs that map to it in the limit) was engulfing most of the input space. In the second test, we fixed the default self-weight at the value that it had at the Stages 2–3 transition and caused all the other weights to change just as they did under delta-rule training. In this case, the default-class influence quickly diminished so that, by the end of the test, no corners were assigned to the default class. These results provide evidence that, under normal training conditions, the growth of the default self-weight is the main cause of the enlargement of the default basin.

The Voronoi boundary and self-weight analyses support the following picture of how the network exhibits non-similarity–based generalization limited to the default class: At the end of Stage 2, the network has formed four, essentially equal-sized attractor basins near the origin stretching out to the indexical bit vectors of the strong classes. In the vicinity of the origin, these basins coincide with the Voronoi compartments defined by the strong-class indexical bit vectors. The default-class fixed point is not stable at this point, but it becomes so at the Stages 2–3 transition and its basin begins to grow. It grows along the Voronoi boundary and spreads out toward the strong bit vectors themselves, encroaching on the strong-class attractor basins. In the process of this encroachment, which is driven by the growth of a single parameter, the default-class self-weight, the pre-image of the default-class attractor encompasses a number of (input space) bit vectors that are more similar to strong classes than to the default class. Therefore, the network exhibits non-similarity–based generalization of the default. The strong-class attractor basins do not grow during this period, but the parameters controlling their structure adjust so that the strong-class exemplars remain within the appropriate basins. Because the strong classes' attractors formed a Voronoi partition of the region around the origin originally, and they do not grow, they never exhibit non-similarity–based generalization.

In the next section, we show that these results allow us to portray the evolution of the network at a level of description that corresponds to that of rule-based models, facilitating careful comparison between the two. This provides a basis for explaining stages, identifying analogs of parameters and the Elsewhere Condition, and clarifying "number of mechanisms."

6. General discussion

6.1. Summary

Recurrent connectionist models and human subjects were both trained to classify feature vectors. There were four "strong" classes and a "default" class. The members of each strong class shared four features that distinguished them from all other exemplars. The members of the default class were heterogeneous—there was no feature pattern common to all of them.

Both models and humans went through three distinct stages on the way to effective performance: In the first stage, they guessed at random; in the second stage, they achieved high accuracy on strong-class exemplars but continued to exhibit low accuracy on default exemplars, incorrectly placing them into strong classes on the basis of similarity; in the third stage, they achieved high accuracy on all training exemplars.

Both the models and the humans were tested on their knowledge of the items and generalization performance after the end of training. Both the models and the humans showed substantial (NSBG) into the default class but relatively little NSBG into the strong classes. This behavior is in keeping with the natural language finding that strong classes are associated with graded properties (like similarity) and default classes are less sensitive to graded properties.

In summary, both the models and humans exhibited two kinds of qualitative differences in the sort that have previously inspired theorists to posit the existence of multiple mental mechanisms: distinct stages of learning and differential sensitivity to similarity.

6.2. Representation questions

We now examine the model's claims to find out how it is similar to and different from symbolic approaches to the phenomena.

6.2.1. Stages

When we speak of stages here, we are concerned with qualitative changes in behavior that take place over the course of a learning session. In the study of development, stages are of central interest, but they are notoriously hard to theorize about because the observables tend to change continuously (so stage boundaries are hard to pinpoint), different individuals exhibit different temporal profiles, and there is no agreed-upon definition of stages. Although connectionist models often provide numerical simulations of such behaviors, in the absence of a theoretical analysis of the networks, it is hard to extrapolate with confidence beyond the observations of the models. Dynamical analyses like those we have described offer some help. As noted above, a bifurcation is a change in a system's topology that accompanies continuous change in a control parameter. In the present case "topology" refers to the number and stability characteristics of the system's equilibria. Bifurcations provide a natural formal model of stage boundaries. We showed, for example, that the model exhibited a Stage 1 (called a "phase" in dynamics) when none of the class vectors was stable; a Stage 2 in which the strong, but not the default vectors were stable; and a Stage 3 in which all were stable. When the model is in each stage, we can be sure that small variations in the state and the parameters will reliably produce the same behaviors at activation convergence.

How would such stages be handled by a symbolic model? A natural assumption is that at Stage 1, no rules exist. The choice-making behavior is thus random. The onset of Stage 2 might then involve the advent of "lexical entries" for the strong-class items. Each entry specifies the (core) features of the item. The assimilation of the default items to the strong classes can be explained by assuming that if an item is presented which does not correspond to a lexical entry, then the entry with the closest feature match is chosen. Like the 2LC model, this system predicts similarity-based assimilation of novel items and lack of non-similarity–based generalization at this stage. Eventually, the symbolic system could adopt a general rule and handle the default cases with this rule. It is not clear, however, that there is any reason the system should adopt a general rule.[12] For example, it seems just as logical that it would memorize a separate rule for each default exemplar. Nor is it clear when it will adopt a default rule. By contrast, in the 2LC model, the advent of the default rule occurs as a consequence of the learning dynamics: The benefits of growing the default self-weight outweigh the benefits of detecting the features of the default elements.

6.2.2. Parameters

The finding of Section 5.2 that a single control parameter (the default-class self-weight) drives the expansion of the default basin is relevant to the way the model generalizes at the end of training. This single parameter cannot selectively expand the basin to capture just the default-class trajectories. It must expand the basin in many directions simultaneously. Therefore, it captures many other input patterns, some of which are nearer to strong exemplars than to default exemplars. In this regard, the self-weight is like a parameter in the Principles and Parameters (P&P) approach to language learning (cf. Chomsky, 1981; Fodor, 2001; Gibson & Wexler, 1994; Manzini & Wexler, 1987): A change in its value can be triggered by a few data points, but the change has implications for many others.

There are two important ways in which the self-weight is different from such a P&P parameter: (a) unlike P&P parameters, which are assumed to be architecturally specified, the function of the default self-weight as a basin-defining parameter is emergent—there are many other environment–network combinations in which a parameter playing this role will not develop (see Kukona, 2011, for an example of different emergent pattern in a 2LC model in a different linguistic environment); and (b) the self-weight parameter, unlike P&P parameters is continuous valued, and the final shape of the basin is a consequence of a delicate negotiation between the error signal coming from the default-class trials, which drives growth of the default self-weight, and the error signals coming from the strong-class trials, which cause the other recurrent weights to adjust in a way that preserves accurate performance on the strong classes.

Indeed, the qualitative form of the generalization varies as a function of subtle variations in the random ordering of the input patterns during the training phase; the structure of the boundary between the default class and the strong classes depends on quantitative, not qualitative features of the behavior. This means that the actual parameterization of the relationship between default and strong classes in the 2LC model is high dimensional even though its general structure is dominated by the one-dimensional default self-weight. The parameters of the P&P framework do not accommodate such subtle quantitative variation—they are either on or off.

6.2.3. MOM versus OM

The question of how to count mechanisms in the 2LC framework is tricky. There is a set of weights for which the entire input hypercube is mapped to the default class. One might say that such a weight set implements a rule with no exceptions (a one mechanism device). There is also a set of weights for which the input hypercube is divided into equal volume, equal shape subsets which each map to one of five classes. One could describe this weight set as implementing pure, similarity-based classification (another one mechanism device). The weights of the current network at the end of training are one point in a high-dimensional continuum that mixes properties of these two extremes. Such a mixture could be interpreted as a more than one mechanism device, but, in the continuum, there are many structurally distinct (in the sense that they make categorically different judgments about novel items) mechanism mixtures. This property is inconsistent with some versions of the MOM hypothesis which assume precisely two mechanisms organized in a dual-route relationship (e.g., Coltheart, Rastle, Perry, Langdon & Ziegler, 2001; Pinker, 1999).

6.2.4. Default/strong asymmetry and the Elsewhere Condition

Our account of the asymmetry between strong and default classes is very close to the view put forth by Hare et al. (1995) (Fig. 1). There are small convex regions of the input space that are mapped to the strong classes, and there is one large region filling in the space between the strong-class clusters that is mapped to the default class. Hare et al. described the regions as attractor basins, although, as noted above, their network does not have attractors of its activation dynamics. The eigenvalue analysis in Section 5 indicates that, in the present case, the indexical bit vectors become attractors of the class network.[13]

Following Hare et al., we can point to a key insight about how our one mechanism model shows qualitatively distinct behavior for strong and default classes. First, the one mechanism property is constituted in the fact that all categories are implemented as attractor basins, which are sets in a vector space. The qualitative distinction in the timing of the emergence of the strong and default categories (i.e., the staging) comes about because the strong classes are easy to detect via the Input → Class weights, but the default category does not form as quickly, either because it is difficult or because it is impossible to classify the default exemplars by adjusting these weights only.[14] Therefore, the strong-class equilibria stabilize first and the default basin develops later, expanding around the Voronoi boundary and approximating their complement. The set-complement relationship corresponds to an Elsewhere Principle. However, unlike the symbolic model, the current model makes principled predictions about how novel forms will be classified and about how the distribution of training forms determines the generalization behavior.

6.3. Future work

An obvious discrepancy between our findings and the research on natural language default categorization phenomena is that we find overirregularization in both humans and models, while the language research has found a preponderance of overregularization (although there is also evidence for overirregularlization, it is less common—Bybee & Slobin, 1982; Xu & Pinker, 1995). We mentioned above that we believe the difference stems from the distributional structure of the training data. Exploratory simulations show that if we introduce a conditioning environment for the categorization scheme (if the pragmatic feature is PAST, then use past inflection; if the pragmatic feature is PRESENT, then use present inflection) and if we increase the frequency of the default classes relative to the strong classes, then the model exhibits a preponderance of overregularization. This observation supports our claim that the distribution matters. Additional paired modeling and category learning work are natural here.

In this study, we have focused on a very simple recurrent network model. One may wonder whether the results extend to other connectionist models. All recurrent connectionist models are dynamical systems and they exhibit stabilities across the range of types that have been identified in dynamical systems research (fixed points, limit cycles, space-filling trajectories, and chaos). We think this bodes well for the possibility of using dynamical systems analysis to clarify the relationship between connectionist and symbolic approaches more broadly. On the other hand, much connectionist research on both categorization and language has focused on feedforward models. These cannot be analyzed in the same way. Nevertheless, there is a clustering structure in the hidden units of a feedforward model that seems to be related to the stability structure in dynamical models (Tabor, 1994; Tabor & Tanenhaus, 1999); this connection deserves further investigation.

In this regard, we have run simulations of the current training results with a feedforward, connectionist category learning model, ALCOVE (Kruschke, 1992)—the model makes very similar predictions to the 2LC model on the task at hand, but the reasons for the stages and generalization are harder to discern. While the 2LC model captures the default category exemplars by mapping them to the Voronoi boundary, ALCOVE appears to capture them via its attentional weighting mechanism (when this is removed, it fails to categorize well). Future empirical work can helpfully distinguish the two approaches by exploring stimuli in which attentional weighting cannot yield elsewhere behavior.

6.4. Conclusion

In sum, consideration of stages, parameters, and the default-strong asymmetry all highlight the ways the 2LC model develops structures that are similar to those invoked in symbolic models, permitting comparison of the models and revealing contrasting predictions. The tools of dynamical systems theory proved helpful in discerning these structures.

Acknowledgments

This material is based on work supported by the National Science Foundation Grant No. 1059662. We thank the members of SOLAB, Reed Helms, Emily Pakstis, Jeff Rodney, Emily Szkudlarek, Kevin Semataska, and Olivia Harold, who helped run the experiment.

Notes

1. In Kiparsky's (1973) formulation for phonology, the two rules must also make the same structural change or incompatible structural changes.
2. Given a set A in another set X, the *complement* of A is the set of points in X that are not in A.
3. Unlike in the strong classes, where the additional features provide no useful information about class membership, the additional features of the default-class items provide some information that could be used, in combination with other default-class features, to distinguish the default class from the strong classes. Even with these extra features, however, there is no set of features that is shared by all default-class exemplars.
4. Thus, Eq. 1b was replaced by $a_i(t+1) = a_i(t) + \frac{1}{\tau_a} net_i \cdot a_i \cdot (1 - a_i)$ and Eq. 2 was replaced by $w_{ij}(t+1) = w_{ij}(t) + \frac{1}{\tau_w}(t_i - a_i) \cdot a_j$.
5. The unit hypercube in two dimensions is the square whose corners are the bit vectors in R^2 (i.e., [0, 0], [1, 0], [1, 1], and [0, 1]). This characterization generalizes to lower and higher dimensions.
6. The preference for Class 3 appears to occur because Class 3 had a feature that was almost a distinctive cue to Class 3 membership (it was shared with only one default exemplar and no strong exemplars). Therefore, in classifying novel items, the network took this feature as a very strong indicator of Class 3 membership. No other strong class possessed such a distinctive feature. The distribution of features across novel items was fairly uniform, so Class 3 had a net advantage. We made the feature distribution heterogeneous in this way in an effort to balance the feature count variance in the strong and default classes.
7. We chose to include a relatively large number of trained default items in the test phase to improve our ability to tell whether participants were maintaining an accurate encoding of the default class during the test. This choice may have introduced a response bias for the default class. The analyses we report below are not concerned with the rate of default responses in comparison to strong responses, so we do not think this possible response bias is problematic for our conclusions.
8. All participants struggled to learn the default class. The difference between the successful and unsuccessful participants was largely that the unsuccessful ones struggled more with the default class and did not meet our criterion for the last three blocks of training. We thus believe our analysis of the highest performing participants is telling us about a system of encoding toward which most, if not all, participants were striving.
9. The Jacobian is the matrix of partial derivatives of the activation change functions with respect to the activation variables (Hirsch & Smale, 1974; Strogatz, 1994).
10. A metric space is a space on which a distance metric is defined. A distance metric on a space X is a map d: $X \rightarrow R$ satisfying $d(x, x) = 0$, $d(x, y) = d(y, x)$, and $d(x, y) + d(y, z) \geq d(x, z)$, for all x, y, z in X.

11. We also examined the stability of other hypercube corners. Except for the origin, which is always non-hyperbolic, the remaining hypercube corners were only fleetingly non-hyperbolic. One of them, the *farthest corner* (the vector of all ones), became stable for the period of strong-class instability at the beginning of training and then switched to being unstable. Many of the other corners became stable along with the indexical bit vectors, but almost all of them then became unstable by the end of training. A non-origin, non-indexical bit corner has more than one bit on at once. In Section 3 above, we treated cases where the trained model converged on such a corner as cases where it chose randomly among the activated classes. As we noted, in the model, this rarely occurred with the training inputs and occurred somewhat more frequently with the novel test patterns. Equilibria of the system can also arise if net_i is zero for some i. Such equilibria lie on the surface of the unit hypercube, but they may not to lie at the corners (i.e., they can have non-extreme activation values). There are many possible fixed points of this sort. We did not attempt to locate them all. However, in our observation, the network almost never converged on an activation value that was not 0 or 1, so non-corner stabilities seem not to play a major role in the dynamics.
12. Default Logic (Reiter, 1980) is a symbolic logic that implements default rules as well as specific assertions. In the mid-late 20th century, default rules were also common in implemented production systems (e.g., Forgy & McDermott, 1977). More recently, Pascal Nicolas and colleagues (e.g., Nicolas & Duval, 2001) have investigated the induction of rule systems in default logic. It will be helpful, in future work, to formally compare this induction perspective to the connectionist mechanism we describe here.
13. Hare et al. motivated their use of a network with hidden units by noting that a network without hidden units can only solve linearly separable problems (Minsky & Papert, 1969). The mapping we study here is not linearly separable. Nevertheless, our two-layer network learns it. This is because recurrently connected units can play a role analogous to that of hidden units by twisting the class space.
14. In the present model, the mapping is impossible to implement directly via the Input → Hidden weights because it is not linearly separable. In other simulations, a qualitative staging contrast develops in cases where the default class is simply difficult (but not impossible) to classify directly.

References

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

Ashby, F. G., Alfonso-Reese, L. A., Turken, A. U., & Waldron, E. M. (1998). A neuropsychological theory of multiple systems in category learning. *Psychological Review*, *105*, 442–481.

Bartos, P. D. (2002). Connectionist modelling of category learning. PhD thesis, The Open University.

Beretta, A., Campbell, C., Carr, T. H., Huang, J., Schmitt, L. M., Christianson, K., & Cao, Y. (2003). An ER-fMRI investigation of morphological inflection in German reveals that the brain makes a distinction between regular and irregular forms. *Brain and Language*, *85*, 67–92.

Berko, J. (1958). The child's learning of English morphology. *Word*, *14*, 150–177.

Brown, R., & Hanlon, C. (1970). Derivational complexity and order of acquisition in child speech. In J. R. Hayes (Ed.), *Cognition and the development of language* (pp. 155–175). New York: Wiley.

Bybee, J. L. (1985). *Morphology: A study of the relation between meaning and form*. Philadelphia, PA: Benjamins.

Bybee, J. L., & Moder, C. L. (1983). Morphological classes as natural categories. *Language*, *59*, 251–270.

Bybee, J. L., & Slobin, D. (1982). Rules and schemas in the development and use of the English past tense. *Language*, *58*, 265–289.

Cazden, C. B. (1968). The acquisition of noun and verb inflections. *Child Development*, *18*, 21–40.

Chomsky, N. (1981). *Lectures on government and binding*. Dordrecht, The Netherlands: Foris.

Clahsen, H., & Almazan, M. (1998). Syntax and morphology in Williams syndrome. *Cognition*, *68*, 167–198.

Coltheart, M., Rastle, K., Perry, C., Langdon, R., & Ziegler, J. (2001). DRC: A dual route cascaded model of visual word recognition and reading aloud. *Psychological Review*, *108*, 204–256.

Cortese, M. J., Balota, D. A., Sergent-Marshall, S. D., Buckner, R. L., & Gold, B. T. (2006). Consistency and regularity in past-tense verb generation in healthy ageing, Alzheimer's disease, and semantic dementia. *Cognitive Neuropsychology*, *23*, 856–876.

Cowell, R. A., & French, R. M. (2011). Noise and the emergence of rules in category learning: a connectionist model. *IEEE Transactions on Autonomous Mental Development*, *3*(3), IEEE.

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

Endress, A. D., & Bonatti, L. L. (2007). Rapid learning of syllable classes from a perceptually continuous speech stream. *Cognition*, *105*, 247–299.

Fodor, J. D. (2001). Setting syntactic parameters. In M. Baltin, & C. Collins (Eds.), *The handbook of contemporary syntactic theory* (pp. 730–767). Oxford, UK: Blackwell Publishers.

Forgy, C., & McDermott, J. (1977). OPS: A domain independent production system language. In *Proceedings of the 5th Joint International Conferences on Artificial Intelligence (IJCAI-77)* (pp. 933–939). Cambridge, MA: Massachusetts Institute of Technology.

Friston, K. (2003). Learning and inference in the brain. *Neural Networks*, *16*, 1325–1352.

Gibson, E., & Wexler, K. (1994). Triggers. *Linguistic Inquiry*, *25*, 407–454.

Gold, E. M. (1967). Language identification in the limit. *Information and Control*, *16*, 447–474.

Hare, M., Elman, J. L., & Daugherty, K. G. (1995). Default categorization in connectionist networks. *Language and Cognitive Processes*, *10*, 601–630.

Hirsch, M. W., & Smale, S. (1974). *Differential equations, dynamical systems, and linear algebra*. San Diego, CA: Academic Press.

Hotton, S., & Yoshimi, J. (2011). Extending dynamical systems theory to model embodied cognition. *Cognitive Science*, *35*, 444–479.

Indefrey, P., Brown, C., Hagoort, P., Herzog, H., Sach, M., & Seitz, R. J. (1997). A PET study of cerebral activation patterns induced by verb inflection. *NeuroImage*, *5*, S548.

Jaeger, T. F. (2008). Categorical data analysis: Away from ANOVAs (transformation or not) and towards logit mixed models. *Journal of Memory and Language*, *59*, 434–446.

Jaeger, J., Lockwood, A., Kemmerer, D., Van Valin, R., Murphy, B., & Khalak, H. (1996). A position emission tomography study of regular and irregular verb morphology in English. *Language*, *72*, 451–497.

Kim, J., Marcus, G., Pinker, S., Hollander, M., & Coppola, M. (1994). Sensitivity of children's inflection to grammatical structure. *Journal of Child Language*, *21*, 173–209.

Kim, J., Pinker, S., Prince, A., & Prasada, S. (1991). Why no mere mortal has ever flown out to center field. *Cognitive Science*, *15*, 173–218.

Kiparsky, P. (1973). "Elsewhere" in phonology. In S. Anderson, & P. Kiparsky (Eds.), *A Festschrift for Morris Halle* (pp. 93–106). New York: Holt, Reinhart, & Winston.

Kruschke, J. K. (1992). ALCOVE: An exemplar-based connectionist model of category learning. *Psychological Review*, *99*, 22–44.

Kruschke, J. K. (2008). Bayesian approaches to associative learning: From passive to active learning. *Learning & Behavior*, *36*, 210–226.

Kukona, A. (2011). Self-organization in anticipatory language contexts: A new view of top-down and bottom-up constraint integration during online sentence processing. PhD Dissertation, Department of Psychology, University of Connecticut.

Kukona, A. (2011). Self-organization in anticipatory language contexts: A new view of top-down and bottom-up constraint integration during online sentence processing. PhD Dissertation, Department of Psychology, University of Connecticut.

Laakso, A., & Calvo, P. (2011). How many mechanisms are needed to analyze speech? A connectionist simulation of structural rule learning in artificial language acquisition. *Cognitive Science*, *35*, 1243–1281.

Manzini, M. R., & Wexler, K. (1987). Parameters, binding theory, and learnability. *Linguistic Inquiry*, *18*, 413–444.

Maratsos, M. (2000). More overregularizations after all: New data and discussion on Marcus, Pinker, Ullman, Hollander, Rosen, and Xu. *Journal of Child Language*, *27*, 183–212.

Marcus, G. F., Pinker, S., Ullman, M., Hollander, M., Osen, T. J., & Xu, F. (1992). Overregularization in language acquisition. *Monographs of the Society for Research in Child Development*, *57* (4, Serial No. 228), 1–178.

Mareschal, D., French, R. M., & Quinn, P. (2000). A connectionist account of asymmetric category learning in infancy. *Developmental Psychology*, *36*, 635–645.

Marslen-Wilson, W., & Tyler, L. (1997). Dissociating types of mental computation. *Nature*, *387*, 592–594.

McClelland, J. L., Botvinick, M. M., Noelle, D. C., Plaut, D. C., Rogers, T. T., Seidenberg, M. S., & Smith, L. B. (2010). Letting structure emerge: Connectionist and dynamical systems approaches to cognition. *Trends in Cognitive Sciences*, *14*, 348–356.

McClelland, J., & Patterson, K. (2002). Rules or connections in past tense inflection: What does the evidence rule out? *Trends in Cognitive Science*, *6*, 465–472.

Mitchell, T. M. (1997). *Machine learning*. New York: McGraw-Hill.

Minsky, M., & Papert, S. (1969). *Perceptrons*. Cambridge, MA: MIT Press.

Nicolas, P., & Duval, B. (2001). Representation of incomplete knowledge by induction of default theories. In T. Eiter, W. Faber, and M. Truszczynski (Eds.), *Logic programming and nonmonotonic reasoning. (Lecture notes on artificial intelligence 2173)* (pp. 160–172). Berlin: Springer-Verlag.

Nosofsky, R. M. (1986). Attention, similarity, and the identification-categorization relationship. *Journal of Experimental Psychology: General*, *115*, 39–57.

Oppenheim, G. M., Dell, G. S., & Schwartz, M. F. (2010). The dark side of incremental learning: A model of cumulative semantic interference during lexical access in speech production. *Cognition*, *114*, 227–252.

Pearlmutter, B. A. (1995). Gradient calculations for dynamic, recurrent neural networks. *IEEE Transactions on Neural Networks*, *6*, 1212–1228.

Peña, M., Bonatti, L. L., Nespor, M., & Mehler, J. (2002). Signal-driven computations in speech processing. *Science*, *298*, 604–607.

Penke, M., Janssen, U., & Krause, M. (1999). The representation of inflectional morphology: Evidence from Broca's aphasia. *Brain and Language*, *68*, 225–232.

Pineda, F. J. (1995). Recurrent backpropagation networks. In Y. Chauvin & D. E. Rumelhart (Eds.), *Backpropagation: Theory, architectures, and applications* (pp. 99–136). Hillsdale, NJ: Lawrence Erlbaum Associates.

Pinker, S. (1999). *Words and rules: The ingredients of language*. New York: Basic Books.

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

Pinker, S., & Ullman, M. (2002a). The past and future of the past tense. *Trends in Cognitive Science*, *6*, 456–463.

Pinker, S., & Ullman, M. (2002b). Combination in structure, not gradedness, is the issue. *Trends in Cognitive Science*, *6*, 472–474.

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

Plunkett, K., & Nakisa, R. C. (1997). A connectionist model of the Arabic plural system. *Language and Cognitive Processes*, *12*, 807–836.

Prasada, S., & Pinker, S. (1993). Generalization of regular and irregular morphological patterns. *Language and Cognitive Processes*, *8*, 1–56.

Regier, T. (1996). *The human semantic potential: Spatial language and constrained connectionism*. Cambridge, MA: MIT Press.

Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence*, *13*, 81–132.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning internal represenations by error propagation. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition: Vol. 1. Foundations* (pp. 318–362). Cambridge, MA: MIT Press.

Rumelhart, D., & McClelland, J. (1986). On learning the past tenses of English verbs. In D. E. Rumelhart, J. L. McClelland, & the PDP Research Group (Eds.), *Parallel distributed processing: Explorations in the microstructure of cognition: Vol. 2. Psychological and biological models* (pp. 216–271). Cambridge, MA: MIT Press.

Shepard, R. N. (1987). Toward a universal law of generalization for psychological science. *Science*, *237*, 1317–1323.

Sperling, G. (1960). The information available in brief visual presentations. *Psychological Monographs: General and Applied*, *74*, 1–29.

Stemberger, J. P., & MacWhinney, B. (1986). Frequency and the lexical storage of regularly inflected forms. *Memory and Cognition*, *14*, 17–26.

Strogatz, S. S. (1994). *Nonlinear dynamics and Chaos*. Reading, MA: Addison-Wesley Publishing Co.

Tabor, W. (1994). *Syntactic innovzation: A connectionist model*. Unpublished doctoral dissertation, Stanford University.

Tabor, W., Juliano, C., & Tanenhaus, M. K. (1997). Parsing in a dynamical system: An attractor-based account of the interaction of lexical and structural constraints in sentence processing. *Language and Cognitive Processes*, *12*, 211–271.

Tabor, W., & Tanenhaus, M. K. (1999). Dynamical models of sentence processing. *Cognitive Science*, *23*, 491–515.

Ullman, M., Bergida, R., & O'Craven, K. M. (1997). Distinct fMRI activation patterns for regular and irregular past tense. *NeuroImage*, *5*, S549.

Ullman, M. T., Corkin, S., Coppola, M., Hickok, G., Growden, J. H., Koroshetz, W. J., & Pinker, S. (1997). A neural dissociation within language: Evidence that the mental dictionary is part of declarative memory and that grammatical rules are processed by the procedural system. *Journal of Cognitive Neuroscience*, *9*, 289–299.

Ullman, M. T., & Gopnik, M. (1999). Inflectional morphology in a family with inherited specific language impairment. *Applied Psycholinguistics*, *20*, 51–117.

Williams, R. J., & Zipser, D. (1989). A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, *1*, 270–280.

Xu, F., & Pinker, S. (1995). Weird past tense forms. *Journal of Child Language*, *22*, 531–556.

Xu, F., & Tenenbaum, J. B. (2007). Word learning as Bayesian inference. *Psychological Review*, *114*, 245–272.