Comments welcome! July 2011.

Phonotactic learning without *a priori* constraints: Arabic root cooccurrence restrictions revisited[*]

John Alderete[1], Paul Tupper[1], Stefan A. Frisch[2]
Simon Fraser University[1], University of South Florida[2]

Abstract. Most constraint-based learning systems do not make learning the identity of constraints operative in grammar a significant part of learning. We motivate and implement a connectionist learning system that does precisely this in the domain of phonotactics. In particular, we develop a multilayer feed-forward network that learns the constraints that underlie restrictions banning homorganic consonants, or 'OCP effects', in Arabic roots. The network is trained using standard learning procedures in connection science with a representative sample of Arabic roots. The trained network is shown to classify actual and novel Arabic roots in ways that are qualitatively parallel to psycholinguistic study of Arabic. Statistical analysis of network behavior also shows that activations of nodes in the hidden layer correspond well with violations of symbolic well-formedness constraints familiar from generative phonology. In sum, it is shown that at least some constraints operative in phonotactic grammar can be learned from data and do not have to be stipulated in advance of learning.

Keywords: subsymbolic learning, connectionism, parallel distributed processing, Optimality Theory, Arabic, cooccurrence restrictions, dissimilation, nature vs. nurture

1. Introduction

Much work in generative linguistics is nativist in the sense that the fundamental mechanisms for computing linguistic processes are assumed to be innate. In Optimality Theory (OT), for example, the building blocks for grammar, well-formedness constraints, are universal and innate ((Prince and Smolensky, 1993/2004), (McCarthy and Prince, 1999)). Cross-linguistic differences are accounted for by reranking these fixed constraints. While it is fairly certain that some aspects of language are innate in humans, it is also far from clear which aspects are innate and which simply evolve in the natural course of language development. Results from a host of different research paradigms have shown that many language processes can be learned directly from the statistical structure of experience ((Elman et al., 1996), (Spencer et al., 2009)), including nontrivial ones like dependencies between nonadjacent elements ((Gomez, 2002), (Newport and Aslin, 2004)). Perhaps at least some of the constraints of OT grammars can be learned from experience too.

In a sense, recent work in computational language learning in phonology anticipates this issue. Initial computational work in OT showed that, with a finite set of fixed constraints, complex linguistic systems can be learned within an OT architecture ((Tesar, 1995), (Tesar and

[*] This report has benefited from the comments and questions put to us by Joe Pater, Terry Regier, and Colin Wilson and those raised at the poster session of the 2009 Cognitive Science Conference and the 2010 Computational Modeling of Sound Pattern Acquisition conference at the University of Alberta. It is supported in part by a standard SSHRC research grant (410-2005-1175) awarded to John Alderete and an NSERC discovery grant awarded to Paul Tupper. Thanks also to Navid Alaei for help in finding and processing the Buckwalter Arabic root list and Reem Alsadoon for answering our questions about Arabic linguistics. Usual disclaimers.

Smolensky, 2000)). Related research paradigms, including the Gradual Learning Algorithm ((Boersma, 1998), (Boersma and Hayes, 2001)) and Harmonic Grammar ((Legendre et al., 1990), (Pater, 2009)), modify how constraint-based grammars predict output forms, but they retain the assumption that the constraints themselves are given in advance of learning. More recently, however, (Hayes and Wilson, 2008) call into question this assumption. In their theory, constraints can be induced from the data by search heuristics that select a small number of highly predictive constraints from a quasi-infinite constraint set. While this approach is used more as an inductive baseline to motivate the introduction of more abstract structures, it is notable in that it makes learning the constraints themselves a nontrivial part of learning.

We seek to continue this line of research by providing an additional mechanism of inducing constraints from data. In particular, we develop a connectionist architecture for learning phonotactic constraints. Below we motivate this cognitive architecture and apply it to the problem of learning root occurrence restrictions, or 'OCP effects', in Arabic. Arabic is chosen because large datasets exist, i.e., root lists and psycholinguistic experiments (Frisch et al., 2000), that enable strong tests of model performance. Also, Arabic exhibits graded phonotactic patterns that make it a good test case for any learning system designed to induce constraints. The principal result reported below is that the graded phonotactic patterns of Arabic consonant phonology can be learned as the gradual tuning of subsymbolic constraints in a connectionist network. Learning of OCP constraints in a connectionist network therefore presents a new way of inducing constraints from data.

The rest of the article is organized as follows. The next section summarizes the Arabic data that we attempt to model, including some exceptional patterns that we document in some detail. Section 3 lays out the theoretical context for our model, including a comparison of connectionist learning models with the contemporary models of learning phonology mentioned above. Section 4 lays out the principal assumptions of our connectionist network, and section 5 presents the learning results. The last section discusses some of the issues raised by the research.

2. Root cooccurrence restrictions in Arabic

A root in Arabic is a discontinuous string of consonants that is interspersed with patterns of vowels to form stems. The number of consonants making up the root can be between two and five, but triconsonantal roots are by far the most common. Roots in a sense specify a narrow semantic field within which actual stems are realized. For example, the triconsonantal root *k-t-b* 'writing' can be interlocked with the pattern for the active participle, *CaaCiC*, to form the noun *kaatib* 'writer'. While standard reference grammars, e.g., (Ryding, 2005), tend to distinguish just these roots and patterns, work in contemporary theories of morphology and phonology has further decomposed some stem patterns into grammatical morphemes of two kinds: (i) discontinuous strings of vowels and, (ii) prosodic templates to which the consonantal root and vocalic morphemes are linked up ((McCarthy, 1979), (McCarthy and Prince, 1990)).

Arabic roots exhibit a phonological pattern in which there is a strong tendency against two adjacent consonants having the same place of articulation. This generalization was first clarified in (Greenberg, 1950) and explored further in ((McCarthy, 1988), (McCarthy, 1994), and (Pierrehumbert, 1993)) with different root lists. The chart below, from (Frisch et al., 2004), organizes Arabic consonants into a set of homorganic natural classes typically assumed in prior work, following the autosegmental analysis of (McCarthy, 1988).[1] We refer to these classes below (excluding the uvulars) as ‘same-place’ classes, because they are not co-extensive with the natural classes defined by major place features. As explained below, there are three separate coronal same-place classes, and uvulars are merged with both dorsal and pharyngeal classes. The rate of cooccurrence of two consonants in a root is quantified as a so-called O/E ratio, or the ratio of observed consonant pairs to the number of consonants that would be expected to occur by chance (Pierrehumbert, 1993). The O/E ratios for sequences of adjacent consonants in a root, i.e., the first two or last two consonants, are shown below in Table 1 from a dataset of 2674 triliteral verb roots compiled originally in (Pierrehumbert, 1993) and based on the Hans Wehr Arabic-English Dictionary (Cowan, 1979).

An O/E ratio of less than 1 indicates underrepresentation in the dataset, as shown in all the shaded cells below for all same-place consonant pairs. Uvulars are also significantly underrepresented when they combine with either dorsals or pharyngeals. For this reason, uvulars are generally assumed to be in both same-place classes. While not as strong an effect, coronal stop + fricative pairs are also underrepresented with an O/E of 0.52. Thus, after merging uvulars with dorsals and pharyngeals, there are six same-place classes in Arabic root phonotactics.

Table 1. Co-occurrence of adjacent consonants in Arabic triliteral roots (from Frisch et al. 2004).

	Lab	Cor Stop	Cor Fric	Dorsal	Uvular	Phar	Cor Son
Labial [b f m]	0.00	1.37	1.31	1.15	1.35	1.17	1.18
Cor Stop [t d tˤ dˤ]		0.14	0.52	0.80	1.43	1.25	1.23
Cor Fric [θ ð s z sˤ zˤ ʃ]			0.04	1.16	1.41	1.26	1.21
Dorsal [k g q]				0.02	0.07	1.04	1.48
Uvular [χ ʁ]					0.00	0.07	1.39
Pharyngeal [ħ ʕ h ʔ]						0.06	1.26
Cor Son [l r n]							0.06

This restriction against same-place pairs is also found in non-adjacent consonants, e.g., the first and third consonant of a triliteral root, but the effect is not as strong ((Greenberg, 1950), (Pierrehumbert, 1993), (Frisch et al., 2004); see also discussion below).

The above data shows that roots that contain two same-place consonants are in general prohibited. However, two identical consonants are commonly found in the second and third consonantal positions in triliteral roots, e.g., *madad* ‘stretch’. Most prior work, and the table above, follow (McCarthy, 1986) in excluding roots with pairs of identical segments in counts of same-place consonant pairs because they assume an analysis in which the second and third consonants are derived in some sense (e.g., by autosegmental double-linking or reduplicative copying) from the same underlying consonant. So the two identical surface consonants do not actually constitute a consonant pair for the purpose of the restriction against homorganic consonants ((Coetzee and Pater, 2008), (Gafos, 1998), (Rose, 2000), (Frisch et al., 2004)). We

[1] Following prior work cited above, glides are excluded from the chart because their unusual phonology makes it difficult to establish frequency generalizations.

follow this work for the sake of concreteness, and exclude identical segments in C2C3 position from the set of patterns that our model is designed to account for.

While the generalization banning homorganic consonants is clearly evident in Table 1, a closer look at the facts of consonant cooccurrence reveals many exceptional patterns that contain particular same-place segments in particular positions. For example, in his original 1950 article, Greenberg notes that, while pairs of pharyngeals and uvulars are in general significantly underrepresented in Arabic, most of the exceptions to this restriction are of the form /χCʕ/, which occur at a rate approaching chance. This example, which is typical of many others, gives additional structure to the description of Arabic phonotactics. While there is an over-arching constraint banning pairs of same-place consonants, there are pockets of consonant pairs that are not as underrepresented as one would expect from a blanket restriction against homorganic consonant pairs. We describe these exceptional patterns below to document this additional layer of phonotactic structure. In section 5, we also use this description to ask if our connectionist network learner is sensitive to this level of phonotactic detail.

Our description draws on the data in the (Buckwalter, 1997) root list. This list contains 4,749 roots, including both triliterals (three consonants) and quadraliterals (four consonants), but we exclude the quadraliterals for comparison with most prior work, which has an exclusive focus on triliterals. The triliteral root list contains 3,823 roots, of which 3,489 are not final geminate roots, i.e., roots of the form XYY. We choose to use the Buckwalter list because it contains both nominal and verbal roots, and also has far more triliterals than the Hans Wehr root list, so it is a more representative sample of the total population of Arabic roots. This choice is important for comparing our model with native speaker judgement data in section 5, because the larger sample is a better approximation of what native speakers are exposed to. The Buckwalter root list transcribed in IPA is available from the authors' websites, together with a set of contingency tables documenting consonant cooccurrence.

Table 2 below lists the counts of all exceptional patterns in the Buckwalter corpus to the homorganic cooccurrence restrictions of Arabic, sorted by the six same-place classes and consonantal position. We exclude examples with identical segments, i.e. roots of the form XXY, XYX. A count is given for both the exceptional pattern and the total number of exceptions in the same position and same-place class. For example, there are 2 roots that fit the pattern /dCt/, where /d/ occurs in C1 position, /t/ in C3, and any other consonant in C2 position. This pattern accounts for 2 of the 15 total number exceptions to the OCP for coronal stops in C1C3 pairings.

Table 2. Exceptional patterns in Arabic triliteral roots, sorted by same-place class and position

	C1C2		C2C3		C1C3			
Labial	fmC	1			bCf	1		
3×3					bCm	9		
					fCm	11		
Totals		*1*		*0*				*21*
Coronal stop	dˤdC	1	Ctd	4	dCt	2	dˤCd	2
4×4			Ctˤd	1	tˤCt	3	dCtˤ	1
			Cdˤd	3	tCd	1	dˤCtˤ	3
					tˤCd	2	dCdˤ	1
Totals		*1*		*8*				*15*
Coronal fricative	sðC	2	Cʃz	1	ʃCθ	2		
7×7	ʃðC	4			ʃCð	1		
	ʃsC	1			ʃCs	3		
	ʃzC	1			ʃCsˤ	1		
	ʃsˤC	2			ʃCzˤ	1		
	ʃzˤC	2						
Totals		*12*		*1*				*8*
Dorsal	gkC	1	Cqg	1	gCk	1	ʁCq	6
5×5	χgC	1	Cgq	2	χCk	1	kCχ	3
	ʁgC	1	Cχq	1	kCg	1	gCχ	2
	χqC	1			qCg	4		
	ʁqC	1			χCg	5		
	kχC	1			ʁCg	1		
	gχC	2			gCq	2		
	kʁC	3			χCq	5		
Totals		*11*		*4*				*31*
Pharyngeal	ʔχC	4	Cχʕ	2	ʔCχ	1	ʔCh	3
6×6	ʔħC	3			ʔCħ	1	χCʔ	5
	ʕhC	4			χCʕ	9	ʁCʔ	1
	ʔhC	2			hCʕ	8	ħCʔ	3
	hʔC	1			ʔCʕ	1	ʕCʔ	1
					ʕCh	5	hCʔ	6
Totals		*14*		*2*				*44*
Coronal sonorant	nrC	1	Crl	2	rCl	14		
3×3	rnC	7	Cnl	1	nCl	22		
			Clr	1	nCr	23		
			Cnr	7	lCn	11		
			Cln	5	rCn	15		
			Crn	9				
Totals:		*8*		*25*				*85*

These patterns support and extend some of the observations made in prior work. For example, if one distinguishes these patterns by their position in the root, the number of attested pairings of non-adjacent same-place consonants (C1C3) comes to 45 (from 204 roots), which far outnumbers exceptional patterns in both initial C1C2 (= 23 patterns from 47 roots) and final C2C3 (=14 patterns from 40 roots) pairs. This fact is consistent with the observation that non-adjacent consonant pairs are less restricted than adjacent pairs ((Greenberg, 1950), (McCarthy, 1994), (Pierrehumbert, 1993)). The exceptions to the constraint against two coronal fricatives also reveals a static generalization that Greenberg mentions in passing, namely that most of these exceptions involve /ʃ/ as an initial member of the consonant pair. Finally, these exceptional patterns also often differ in whether they are the lone examples in an otherwise exceptionless pairing of same-place consonants, or they are instead one exceptional pattern among many. For example, there is just one exceptional pattern to the restriction against two pharyngeals in C2C3 pairs, /Cχʕ/, but there are 5 distinct patterns in C1C2 pairs and 12 in C1C3 pairs. The facts above give richer structure to Arabic phonotactics, and we use this as a way of testing how well our learning model has learned the restrictions on homorganic consonants.

We can summarize these facts with some guiding assumptions from Autosegmental Phonology. The Obligatory Contour Principle (OCP; (Leben, 1973), (Goldsmith, 1976), (McCarthy, 1986)), and extensions of it in Optimality Theory ((Myers, 1997), (Suzuki, 1998)), provide a means of formalizing specific consonant cooccurrence restrictions described above ((McCarthy, 1988), (Yip, 1989), (Padgett, 1995)). OCP-Place constraints effectively ban two segments that have identical specifications for the major place features, e.g., OCP-Labial bans a form with two labial segments. In many languages, as in Arabic, additional 'subsidiary' features are needed to further specify the set of features that must be identical. Thus, (Padgett, 1995) argues for three sets of OCP-Coronal constraints in Arabic, OCP-[Coronal, +son], OCP[Coronal, -son, -cont], OCP[Coronal, -son, +cont], to account for the basic fact that the same-place classes of Arabic subdivide the coronals into three classes: sonorants, stops, and fricatives. To these, we add OCP[labial], OCP[dorsal], and OCP[pharyngeal], to cover the course-grained constraints exhibited in the O/E patterns in Table 1.

In addition, two other factors are important: proximity, because non-adjacent consonants have weaker restrictions, and phonological similarity. In Table 1, for example, we see that there is a stronger restriction of two coronal stops than a coronal stop and fricative, because in the latter case, the segments differ in continuancy and are therefore less similar phonologically. In section 5, we review psycholinguistic evidence of the fact that phonological similarity is a factor in native speakers' intuitions about these restrictions. Finally, while we do not know if native speakers have robust intuitions about the exceptional patterns in Table 2, we may also ask if our analysis can discriminate these fine-grained exceptions, essentially on a segment-by-segment basis, to the OCP constraints.

3. The context of connectionist grammars in generative phonology

Connectionist networks (c-nets) compute input-output processes by sending information through a web of simple processing units. C-nets are often said to be neurally-inspired, but connectionist researchers generally do not claim that c-nets are entirely brainlike in the way they process information, and nor do we. The important point is that providing a model of this micro-structure constitutes a theory that makes new predictions, which we explore below in the context of the problem of constraint induction.[2]

Information is passed through a c-net by computing the activation states of simple processing units. The flow of this information is neurally-inspired in the sense that the computation of these states is done in parallel, and the activation state of one unit is affected by the activation state of other units connected to it in the network. As shown below, units are often organized into distinct layers corresponding to different types of representations. For example, input and output layers are distinguished from the hidden layer, an internal representation that restructures the input representation as a function of the first set of connection weights. The activation state of any particular unit is a function of the weighted sum of the activation states of all units sending information to it. Thus, in the enlarged fragment of unit m on the right of Fig. 1, m's activation is the weighted sum of the activation values from all units sending information to m, transformed by an activation function. A commonly used activation function is the sigmoid logistic function, which has the effect of squishing the sums of input activation states into a fixed range.

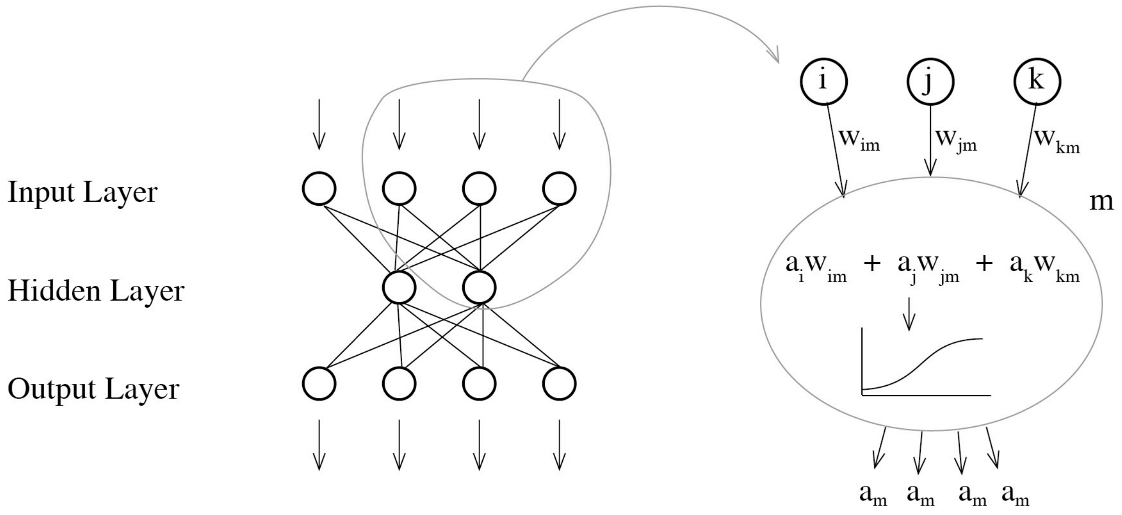

Figure 1. Information processing in a multilayer network.

C-nets and their training are characterized by a host of additional parameters (e.g., bias units that establish thresholds for activation states, plasticity parameters in adjusting connection strengths, etc.) and assumptions about the overall network architecture (number of layers, nature of representations, feed-forward vs. recurrent), and we flesh out these parameters for our c-net below in section 4.

Connectionist grammars are information processing models that take inputs and generate outputs, and so they can be compared with symbol-manipulating grammars as generative models. The well-known analysis of the English past tense in (McClelland and Rumelhart, 1986), for example, generates past tense forms from English present forms by computing representations for inflectional morphology through a multilayer node network. C-nets have also been developed

[2] See ((McLeod et al., 1998), (Mitchell, 1997), and (Thomas and McClelland, 2008)) for more thorough introductions to connectionist networks that go beyond this crisp overview.

that capture the facts of traditional problems in phonology. For example, (Hare, 1990) designed a sequential network to capture some of the core facts of Hungarian vowel harmony. Hare showed that by using a context layer, which in a sense remembers the structure of preceding elements ((Jordan, 1991), (Elman, 1990)), the c-net could explain the effect of similarity and proximity on vowel harmony, i.e., the fact that the more similar and closer the target and trigger are, the stronger assimilatory effect; see also (Wayment, 2009) on attractor networks producing similar effects for a wider set of facts. Another example is the c-net developed in (Legendre et al., 2006) to account for the now classic OT analysis of Tashlhyt Berber syllabification (see (Prince and Smolensky, 1993/2004)); see also (Smolensky et al., To appear) for the next generation of connectionist models that resolves some of the problems of this particular approach. We sketch the model of this last example, dubbed `Brbrnet`, in order to introduce some of the parallels between micro- and macro-structure learning below.

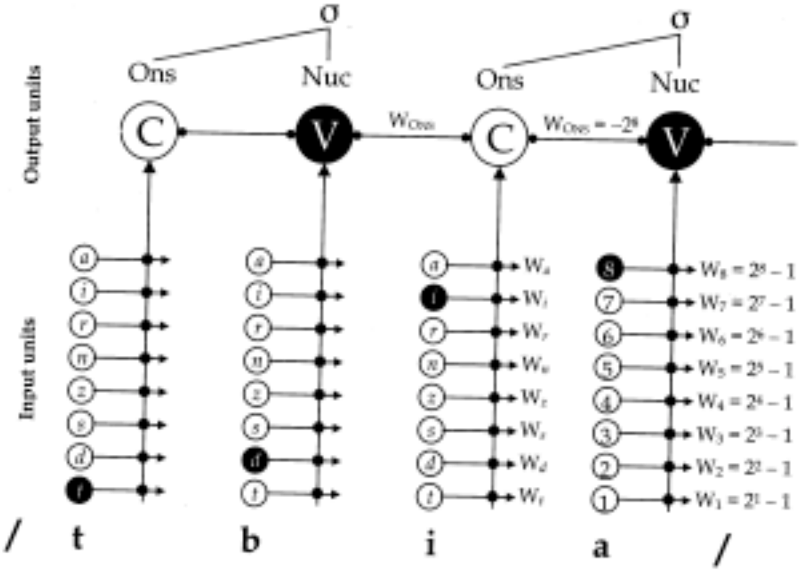

Figure 2. Brbrnet: Syllabifier for Imdlawn Tashlhiyt Berber (Legendre et al., 2006)

The input to `Brbrnet` is a sequence of segments comprising a word. Each segment is encoded in terms of its sonority, which is fed into the network as the activation state of an input node. `Brbrnet` computes the syllabification of these segments as a pattern of activation across an output layer. This output layer consists of a string of nodes each linked to a corresponding input node by an excitatory connection (=positive connection weight). Each output unit is also linked to the output nodes adjacent to it by an inhibitory connection (=negative connection weight). `Brbrnet` is a recurrent network in the sense that the output nodes are fed back onto themselves until a stable state is reached. Output nodes thus compete with each other to determine syllabic roles such that the closer an output node is to being active (= syllable peak), the greater the pressure on adjacent nodes to not be active (= syllable margin).

Besides illustrating complex phonological behavior at the micro-structure level, this example provides a way of distinguishing the symbolic constraints of standard OT and subsymbolic constraints of c-nets. The symbolic constraints of OT express some requirement on output form and inspect outputs for how many times this requirement is violated. Onset, for example, is a symbolic constraint that requires syllables to begin with consonants.

One can also view the dynamic computation of activation patterns in a c-net as constraint satisfaction, but satisfaction of subsymbolic constraints rather than symbolic constraints ((Smolensky, 1988), (Smolensky and Legendre, 2006b)).[3] Subsymbolic constraints are defined as the connection weights between nodes. A subsymbolic constraint can be a single connection, or sets of connections within the larger node network. If the connection between two units is positive, the unit sending information tries to put the unit immediately downstream into the same positive state it is in. The constraint is in a sense satisfied if the state of the receiving node resembles the state of the sending node. If the connection is negative, the sending unit tries to put the receiving unit in the opposite state, so negative weights are satisfied by inducing opposite activation states downstream. Like constraint satisfaction in OT, not all constraints can be satisfied in c-nets. But as activity flows through the network, or cycles through recurrent networks, the activation values of individual units will change in a way that better satisfies these positive and negative constraints. This is the principle of Harmony Maximization of ((Legendre et al., 1990), (Smolensky and Legendre, 2006a)).

The analogue to the symbolic constraint Onset in `Brbrnet` is the set of all connections between output nodes in Figure 2 (Legendre et al., 2006). These connections are negative numbers, so, over time, as certain output nodes get higher activity values, they push their neighbors into states with low activity, i.e., they force them to be margins. Subsymbolic constraints are therefore not global assessments of some property of a macrostructure representation, like whether the syllable initial position is filled with a consonant. They are the combined effect of microstructure links that can be scattered across the network. This has important consequences for constraint induction, because the problem of 'learning the constraints' is characterized more precisely as a problem of learning the correct configuration of connection weights.

Contrast this analysis of the learning problem with contemporary approaches to learning constraint-based grammars. In most constraint based learning systems, the function computed by a constraint is not learned at all. Instead, its importance in the grammar, i.e., its rank in a constraint hierarchy or its weight in the system, is learned in isolation. For example, in Harmonic Grammar, (Coetzee and Pater, 2008) model the learning of cooccurrence restrictions similar to the ones discussed here using a standard gradient descent algorithm for learning constraint weights. In particular, they provide their learner a set of 24 symbolic constraints on feature cooccurrence that prohibit consonant pairs with the same Place specification and also match on other features. The weights of these 24 constraints, plus faithfulness for feature realizations, are gradually adjusted in response to errors. Thus, like many other contemporary models of learning constraint-based grammars, e.g., Multi-Recursive Constraint Demotion ((Tesar, 1995), (Tesar and Smolensky, 2000)) and the Gradual Learning Algorithm ((Boersma, 1998), (Boersma and Hayes, 2001)), the actual functions computed by grammatical constraints are not part of learning.[4]

A recent notable exception to this state of affairs is the Maximum Entropy (MaxEnt) model for learning phonotactics (Hayes and Wilson, 2008). This model, while it shares some important

[3] We avoid the distinction between hard and soft constraints often used in comparing connectionist networks and symbolic-computational systems because hard constraints are often understood as unviolated constraints. OT uses symbolic constraints that are not hard constraints in this sense, because they can be violated if this leads to satisfaction of a higher ranking constraint.

[4] But see Pater (2009) for discussion anticipating this issue and a conjecture on how constraint induction can be guided by principles of phonetic grounding.

properties with Harmonic Grammar, includes a mechanism for selecting the constraints operative in the target grammar by using some well-known search heuristics in machine learning, i.e., the accuracy and generality of a constraint. Given input data, and given a minimal set of assumptions about the form of phonotactic constraints (i.e., SPE feature matrices and constraint schema for banning and requiring combinations), the search heuristics select a set of highly predictive constraints from a vast universe of well-formedness constraints.

This selection mechanism generates what Hayes and Wilson call an 'inductive baseline', or a minimal constraint system derived essentially from the data. The inductive baseline is used primarily in (Hayes and Wilson, 2008) as a tool for theory comparison and motivating certain kinds of theoretical assumptions, like nonlinear tone and prosodic structure. We agree that identifying an inductive baseline is a useful way of identifying problem spaces and show in section 6 how c-nets can produce a different kind of inductive baseline. We also believe that the success of Hayes and Wilson's approach in cases like English onsets provides additional motivation for exploring it as a central theory of where well-formedness constraints might come from. In their study of English onsets, for example, search heuristics selected 23 phonotactic constraints. When weighted in the larger MaxEnt grammar, the resulting grammar predicted grammaticality judgments that correlate rather well with behavioral data on English. Indeed, Hayes and Wilson show that their induced constraints fare as well as five other contemporary models of English onsets in which the grammatical constraints were given in advance. In other words, for English onset phonotactics, it is hard to find a better grammar than a MaxEnt grammar derived from data.

We accept that this style of constraint induction has its limits, but given this success, we choose to explore the idea that certain constraints in phonotactic systems could emerge in the natural course of language learning. Connectionist networks have a number of properties that make them suitable for this kind of investigation. First, a number of learning protocols exist for modeling learning as the gradual adjustment of connection weights. We illustrate below with standard backpropagation learning (Rumelhart et al., 1986a) how knowledge of the phonotactic system can be gradually accrued through incremental exposure to language forms. Second, many phonological generalizations are gradient in nature, characterized both by their distributions in lexicons and native speaker intuitions of these generalizations. C-nets are capable of describing gradient trends in the data because of the continuous nature of the functions they compute ((Rumelhart et al., 1986b), (Bybee and McClelland, 2005)). Third, c-nets are sensitive to fine-grained similarity structure (Rumelhart et al., 1986b), which is again indispensible to the analysis of language which is often sensitive to similarity structure. Fourth, c-nets have been shown to extract over-arching generalizations in tandem with well-defined exceptions to these generalizations, e.g., (McClelland and Rumelhart, 1986). Finally, as demonstrated above, c-nets can be conceptualized as constraint-based systems that optimize over a set of subsymbolic constraints embedded in the c-net. It is therefore possible to scrutinize the performance of the network and analyzing the behavior of specific constraints in the model. All of these properties are relevant to analyzing Arabic root phonotactics ((Greenberg, 1950), (Frisch, 1996), (Frisch et al., 2004)).

In the context laid out above, we establish that a connectionist learning system can 'learn the constraints' by learning the correct configuration of connection weights. This assumption should not be confused with the characterization frequently given to connectionist approaches to cognitive processes, namely that c-nets are a complete 'blank slate' that is completely free of bias and *a priori* assumptions. Clearly, there are some assumptions that we make about the initial state of learning, described in detail below, and there are also assumptions about the model that do not change in response to data. For example, we assume a default set of initial values for connection weights, and specific plasticity and damping parameters relevant to learning. We also assume a fixed number of hidden layer units, though we vary this number in testing to find the right number of hidden layer nodes for the data. Furthermore, we use a set of phonological features in representing input forms, which constitutes a substantive hypothesis about the range of natural classes the model can be sensitive to after training. Most of these assumptions are operational in nature and we think that they do not affect model performance enough to matter for our argument. The number of hidden layer nodes is crucial, however, because the right number is necessary to force the network to make the right generalizations. While it is true that we do not make the number of hidden nodes a part of the learning problem in this study, we very easily could, because there are known protocols for pruning and sprouting hidden layer nodes in c-nets (Mitchell, 1997).

The larger point, however, is that the range of possible constraints for assessing consonant combinations is largely free. For any given set of connections in our network, the set of possible constraints these connections can compute is uncountably infinite. This is because the connection weights are assigned on a continuous scale, so, given that subsymbolic constraints are (sets of) connections, there are an infinite set of constraints that are expressible in our network. The open range for constraint definition in this model therefore makes the problem of learning the constraints a non-trivial one.

4. A connectionist model for Arabic root cooccurrence restrictions

The principal goal of this work is to develop a connectionist network that, after training, has induced the constraints that characterize the OCP constraints of Arabic roots. The assumptions about the larger cognitive architecture are specifically tailored to this goal, which we flesh out in detail below.

4.1 Functional overview of the network

The c-net grammar/learner is composed of two modules, an Autoassociator module and an Assessor module, as depicted in Figure 3. The Autoassociator is a single layer c-net that constitutes a simplified production module. It takes as input a triliteral root and attempts to output an identical root. Like human language production, the Autoassociator is noisy in the sense that random variation in the activation patterns may cause it to produce non-identical roots by replacing some or all of the consonants with another Arabic consonant. The output is therefore either a root identical to the input, or a non-identical root that may accidentally correspond to another attested Arabic root or is not an attested root.

The role of the Autoassociator in learning is like the role of production modules in many constraint-based learning systems. In constraint-ranking algorithms in OT ((Tesar and Smolensky, 2000), (Tesar, 2004)), for example, Production-Driven Parsing produces forms that are either correct or incorrect with respect to the target grammar. These incorrect forms, or ‘errors’, are important evidence in grammar learning because the learner compares these errors with target forms and uses this comparison as a way of revising the grammar. The Autoassociator plays a parallel role: it generates forms that can be used as evidence in grammar learning, which we outline in detail below.

It should be said that the Autoassociator does not constitute a realistic psycholinguistic model of speech production. For example, its errors do not have the same structure as human speech errors, nor do they occur at the same frequency (see for example (Goldrick and Daland, 2009) on the error structure and phonological markedness). It is rather an algorithm that transforms actual forms of the target language and supplies these outputs as evidence to a learning system. In this way, the Autoassociator plays a role that is similar to production systems in other learning systems, and it should be evaluated as such.

The Assessor model takes a triliteral root as input and assesses it by assigning an acceptability score ranging from -1 to 1. The acceptability score is a gradient measure of the overall well-formedness of the root, and so the Assessor is a grammatical model in the sense that it classifies input forms for acceptability. As an analogy, the Assessor can be compared to a model of truth-functional semantics that evaluates sentences and facts about the world and assigns a number 0 or 1 to the sentence as its Boolean-valued interpretation (see, e.g., (Ramsey et al., 1990), who developed a connectionist network that computes exactly this type of function). The output of the Assessor, or the activation state of the final output node, is comparable to the relativized acceptability score of Harmonic Grammar (Coetzee and Pater, 2008) and the maxent values of MaxEnt Grammar (Hayes and Wilson, 2008).

The Assessor only makes sensible classifications of the data when it has been trained, which requires the interface depicted in Figure 3 between the two modules. The training regime is described in more detail below, but in essence, the Assessor trains on the output of the Autoassociator. If the Autoassociator gives the Assessor a form that coincides with the input to the Autoassociator (and is therefore an attested root), all connection weights and biases in the Assessor module are adjusted such that the Assessor gets closer to outputting a ‘1’. If instead the Autoassociator gives the Assessor a form that differs from the input to the Autoassociator (usually, but not always, an unattested root), then all connection weights and biases in the Assessor module are adjusted such that the Assessor gets closer to outputting a ‘-1’.

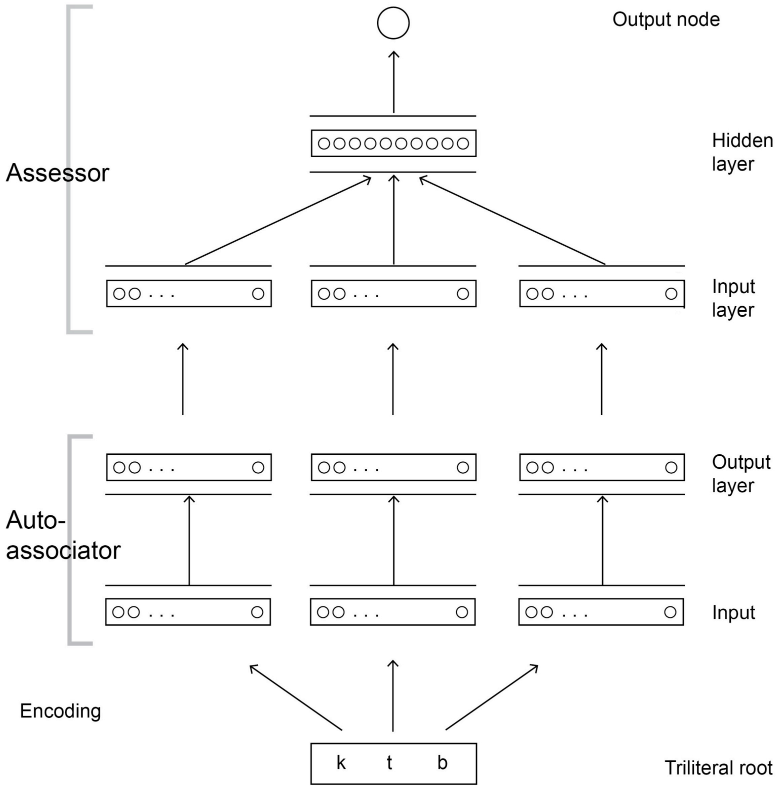

Figure 3. A feed-forward error-generating production module (Autoassociator) and a phonotactic learner (Assessor).

The roots are represented in the two modules using two different representational schemes. The Autoassociator uses a so-called 'localist' representation, meaning that, for each consonantal slot, there is a single active unit that represents that consonant of Arabic in the input representation. For example, when C1 is /b/ the first unit of a sequence of 28 units is '1' and all others are '0'. Though there are arguments for using localist representations for problems like the representation of concepts (see e.g., (Bowers, 2009)), our motivation is purely operational. The output of the Autoassociator needs to provide unattested but logically possible roots in Arabic. Local encoding gives the required output control because the output layer can only represent roots with Arabic consonants. Alternative representation schemes do not give us this control.

The Assessor module, on the other hand, necessarily uses features-based distributed representations, rather like distinctive feature representations commonplace in generative phonology. Distributed representations are simply nonlocal representations. This means that information may be distributed across the representation (= the string of units for a consonant), and the one-to-one relationship between unit activation and consonant type in localist encoding is not guaranteed. Distributed representations are required in the Assessor module because the goal of this module is to capture feature-based generalizations about consonant cooccurrence restrictions. It is simply impossible to achieve this goal unless the activation states of units correspond to phonological feature values, values that are shared among consonants in a natural class. These assumptions therefore require that the interface between the Autoassociator and the Assessor have a conversion process that takes the locally encoded Autoassociator root and converts it to the equivalent distributed representation. This is a simple conversion process and has no function other than making it possible for the Autoassociator to 'talk to' the Assessor.

One aspect of this model that is different from some constraint-based models that use symbolic constraints, e.g., classical OT and Harmonic Grammar, is that the module responsible for language production (Autoassociator) is separate from the module responsible for calculating well-formedness judgments (Assessor). In classical OT, for example, an OT grammar is both a production module in generalized sense (see discusson of Production-Driven Parsing above) and an analysis of the phonotactics.[5] One might reasonably object, therefore, to the overall structure of our model for this reason on the grounds that models that unify these two functions are more parsimonious.

Two points can be made here. First, the separation of production and assessment of phonotactics is more a matter of convenience here than an assumption required on principled grounds, as our real focus is on studying the learning of phonotactics at the micro-structure level. It seems plausible that the two modules could be unified in a revised model. Second, it may be the case that there are empirical reasons for separating the two modules functionally, as argued in Hayes and Wilson (2008) for MaxEnt grammars. These authors draw attention to the existence of alternations that are not phonotactically motivated, which suggests the existence of a module for learning alternations that is distinct from phonotactics. In addition to this evidence, (Zamuner et al., 2006) argue for the functional separation of phonotactics and alternations on the basis of a 'reverse-wug' test done with phonotactically-aware Dutch children. When asked to form singulars from novel plurals like *sladen*, experimental participants do not reply with *slat*, which would imply knowledge of phonotactics. Instead, they have more difficulty in general with such singular-plural pairs when compared with pairs like *slaten/slat* that do not require devoicing. Thus, while our c-net model does have a production module separate from the phonotactic module, this is mostly a matter of convenience and also consistent with the rather preliminary understanding of this issue in the literature.

4.2 Assessor architecture

The next two subsections flesh out the details of the network architecture. The Matlab program that we developed to implement the network is also available on the authors' webpages for further scrutiny and extension to new datasets. The Assessor is a feed-forward neural network with an input layer consisting of 51 nodes, a hidden layer consisting of 5 nodes, and an output layer consisting of 1 node (Figure 3). Every node in the input layer is connected to every node in the hidden layer, and every node in the hidden layer is connected to the output node. Each of these connections has a weight associated with it. The output node and each of the hidden nodes has a bias. These weights and biases are what is modified when the Assessor is being trained (details below). The input layer of 51 units represents a sequence of three consonants, i.e., a triliteral root, as a phonological-feature based distributed representation. We used the phonological features of (Frisch et al., 2004) for Arabic, which is essentially a variant of the widely used feature set from (Clements and Hume, 1995), but adapted for Arabic. The properties of this system relevant to Arabic consonant phonology are (i) it uses primary place features,

[5] While classical OT simply classifies input-output mappings into two categories, 'grammatical' and 'ungrammatical', which cannot directly characterize gradient patterns, the use of the notion of relative harmony in Harmonic Grammar supports the calculation of gradient acceptability scores by subtracting the harmony of some representation from the harmony of its most harmonic competitor (Coetzee and Pater, 2008).

[labial], [coronal], [dorsal], and [pharyngeal], to designate the places of articulation in Arabic, and, like most prior work, (ii) uvulars are both primary [dorsal] and [pharyngeal], because they are restricted in both of these major place classes. There are 17 phonological features in total, so, since each node encodes a single feature value, a sequence of three consonants can be represented with 51 nodes.

Unit activation states correspond to traditional feature values in the following way: '+' = +1, '-' = -1, and all trivially redundant features, i.e., features not specified for a particular segment, receive a 0. These input activations, the weights on the connections between the input layer and the hidden layer, and the biases on the hidden nodes, determine the activation of the hidden nodes. Then, the activation of the hidden nodes, together with the weights on the connections between the hidden layer and the output node, and the bias of the output node, determine the activation of the output node. The computation of activation states through the network is calculated as show below.

(1) Activation in the Assessor module

Let inp_i indicate the activation of the input nodes for $i = 1,...,51$, h_i indicate the activation of the hidden node for $i = 1,...,5$, and out indicate the activation of the output node. The relation between these activations are:

$$h_i = \sigma\left(\sum_j W_{1,ij} inp_j + b_i\right)$$

$$out = \sigma\left(\sum_i W_{2,i} h_i + b_{out}\right)$$

where

$W_{1,ij}$ is the weight on the connection between the jth input node and the ith hidden node

$W_{2,i}$ is the weight on the connection between the ith hidden node and the output node

b_i is the bias on the ith hidden node

b_{out} is the bias on the output node

σ is a sigmoid logistic function with $\sigma(-\infty) = -1,\ \sigma(\infty) = 1$.

4.3 Autoassociator architecture

The Autoassociator is a feed-forward network with no hidden layer (Figure 3). There are 84 input nodes and 84 output nodes (=3 slots × 28 consonants), and each input node is connected to all output nodes with some weight. Each output node has a bias, and also receives a random Gaussian input that is regenerated every time a new root is input to the network. Because of this noise, the Autoassociator does not give the same output each time a fixed input is given, even when the weights are held constant. When a root is input to the Autoassociator, the output nodes are activated. Due to the noise in the system, the output activations are not all 1 or 0, as would be required for the output to be a root itself. The output is therefore converted to a root by taking the largest activation value for each of the three consonant slots and choosing the consonant corresponding to highest activation value.

The output activation of the ith output node of the Autoassociator is given below.

(2) Output activation patterns in the Autoassociator

$$out_i = \sum_j W_{a,ij} inp_j + b_i + \eta \times rand_i$$

where

$W_{a,ij}$ is the weight on the connection between the jth input node and the ith output node,

b_i is the bias on the ith output node

$rand_i$ is a random number drawn from a standard normal distribution each time the output is computed

η is a fixed parameter which specifies the amount of noise in the network. We chose η=0.325. This value yielded a mature network that gave an attested root about half the time and an error root about half the time. We have chosen an error rate of 50% for the Autoassociator for reasons of convenience only. Similar results can be obtained for a much lower error rate at the cost of running for more epochs and making the rate of Assessor training greater for errors than for non-errors. This latter modification would reasonable assuming that the fewer the errors the learner is exposed to, the more salient they are.

The vector *out* is not in the same format as the input vectors, since its elements will not typically be either 0 or 1. The output is converted to the input form by, for each group of 28 nodes, selecting the most active and setting its activation to 1, and then setting all other nodes in that group to zero activation. This procedure could be implemented by introducing inhibitory connections between the output nodes belonging to the same consonant slot, but we have chosen this simpler idealization. This result of this rounding procedure is dubbed *roundout* below.

4.4 Training

The Autoassociator is trained using the Delta rule, a standard method in machine learning for simple one-layer networks (McLeod et al., 1998). The input is randomly selected from the list of attested roots. The network computes an output from the input activations and the randomly generated values $rand_i$ as described above. The output is then compared to the input. The weights and biases are modified based on the difference between the input and the output, as shown below.

(3) Delta rule for updating the Autoassociator weights and biases

$$b_i = b_i + \delta \times \left(roundout_i - inp_i \right)$$

$$W_{a,ij} = W_{a,ij} - \delta \times (out_i - inp_i) \times inp_j$$

The effect of the Delta rule is to change the weights and biases so that the actual output given the particular input is closer to the target output. In our simulations, the variables in b and W_a were all initialized to 0. The training consisted of 10^5 epochs. In each epoch an attested root was randomly selected from the root list and input to the network. $rand_i$ was generated for each node and *out* and *roundout* were computed. The expressions above were used to update b and W_a. Additionally, the weights were then reduced by a factor of $1 - \alpha\delta$ with every epoch, where α= 0.001. This is a standard technique to prevent overfitting by the network (Hastie et al., 2009).

Finally, δ was chosen to vary with time so that δ was 0.1 at the beginning of the training and 0.0001 at the end of the training.

Once the Autoassociator was trained, the Autoassociator weights and biases were fixed and this module was then used to generate training data for the Assessor. The Assessor weights and biases were initialized to small random values before the training. Training consisted of 10^7 epochs. At each epoch of the training the Assessor, a root was randomly selected from the list of attested roots. The root was then passed through the Autoassociator to generate an output root. If the input root and the output root were identical, the target was chosen to be 1. If the input root and the output root were different, the target was chosen to be -1. The output root was then input to the Assessor. Based on the difference between the output from the Assessor and the target, the weights and biases of the Assessor were updated using one step of backpropagation. Backpropagation is a method for training feed-forward neural networks with one or more hidden layers (Hastie et al., 2009). For our network, backpropagation first updates W_2 and b_{out} exactly as in learning by the Delta rule, with the goal of making the output closer to the desired target. Additionally, the activations of the hidden nodes are modified, also with the goal of bringing the value of the output closer to the target. Finally, W_1 and b are modified as with the Delta rule, so that the actual activation of the hidden nodes with the given input is closer to the new hidden node activations. As with training the Autoassociator, there is a parameter δ that controls the rate of learning, which was varied from δ=1 at the beginning of the training to δ=0.1 at the end. To reduce overfitting, weights and biases were decreased by a factor $1 - \alpha\delta$ with every step, where $\alpha = 10^{-5}$.

5. Results

We describe below the performance of the model after training by comparing the Assessor's output to psycholinguistic data (5.1), presenting statistical analyses of the behavior of the hidden layer nodes (5.2), and evaluating how well the Assessor module captures exceptional patterns (5.3).

5.1 Statistical effects of the OCP on acceptability

Because of its design, the trained Assessor module rates triliteral roots in a way that can be compared to human ratings of acceptability. Before we compare Assessor ratings to the judgement data from (Frisch and Zawaydeh, 2001), we give an overview of the module's performance by showing how it classifies all possible roots. Figure 4 below illustrates the rating results for an Assessor network with five hidden layer units, given input trained on the Buckwalter root list (see section 2). The first observation to make is that, while the ratings for the actual roots (a) overlap with the ratings for all possible roots (b) ($n = 28^3 = 21{,}952$), the ratings for the middle 50% of the actual words is centered around the top of the middle 50% of the ratings for all possible roots (compare the first and second boxplots), indicating that network has learned to assign higher scores to the words that it has been exposed to. Second, the opposition between the third (OCP compliant roots (c), from Frisch et al's Experiment 1) and fourth boxplots (OCP violating roots (d), same source) shows that the network has effectively learned the OCP. The middle 50% of the ratings for the OCP compliant roots is well above the middle 50% of the ratings for the OCP violating roots. Thus, when we look at OCP-Place restrictions globally, the Assessor has a strong tendency to rank roots that violate the OCP lower than those that do not.

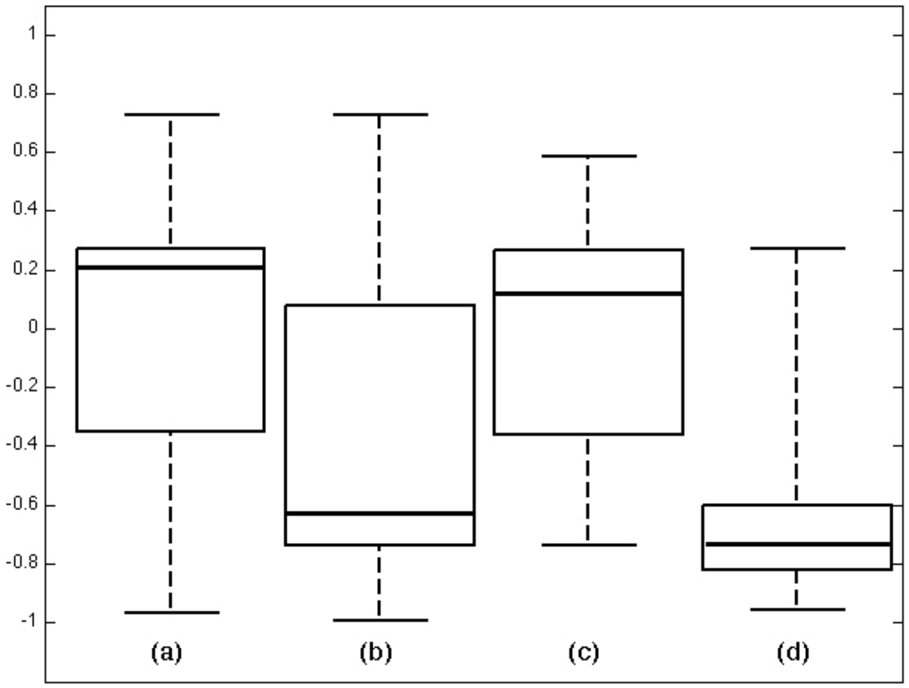

Figure 4. Acceptability scores for one trial of Assessor module with five hidden nodes. Box plots indicate minimum, first quartile, median, third quartile and maximum scores for (a) all attested roots, (b) all possible roots, (c) OCP compliant roots, (d) OCP violating roots.

We can now examine the Assessor's performance by comparing the Assessor's output for the same data presented to experimental participants in (Frisch and Zawaydeh, 2001). This is a strong test of the model because we can perform the same statistical tests and examine the effect of the same factors considered in Frisch and Zawaydeh's study. To do so, we must first summarize briefly the design and results of their study.

Frisch and Zawaydeh (2001) used a wordlikeness experiment to probe the psychological reality of the OCP with 24 native speakers of Jordanian Arabic. Native speaker participants were given a set of inflected nonsense words containing triliteral roots that were manipulated for the following variables: number of OCP violations, expected probability, neighborhood density, bigram probability, and phonological similarity of OCP-violating consonant pairs.[6] Participants were asked to rate words on a seven-point scale for overall acceptability as a word of Arabic, which was the dependent measure on all experiments. The larger finding was that the OCP had a significant effect on subjects' ratings that in general could not be attributed to these lexico-

[6] The terms used in the experiment are defined as follows. 'Expected probability' (abbreviated exp.prob.) is the probability of independent combinations of the segments that make up a form, given their frequency in the lexicon; it is the product of monogram probabilities when more than one segment is considered. 'Neighborhood density' (density) in triliteral roots is the number of existing roots that share two of the three consonants in the appropriate serial positions. Bigram probability with respect to a given two locations in the root is the number of existing roots that have matching consonants in those locations. Similarity of two phonological segments is defined in (Frisch et al., 2004) as the number of shared natural classes over the sum of shared natural classes plus the non-shared natural classes.

statistical effects or accidental gaps. Furthermore, subjects' ratings of these words did fall on a gradient that correlates with featural similarity of the two consonants, as shown in the snapshots of the three experiments below.

Summary of Results of Frisch and Zawaydeh 2001.

Experiment 1. Is the OCP psychologically real, and not just an effect of lexical statistics?

- independent variables: OCP violations, expected probability, neighborhood density
- results/conclusion: significant effect of OCP found on wordlikeness ratings, no other effects found and no interactions; OCP accounts for approximately 30% of subject variability

Experiment 2. Do subject ratings distinguish between systematic gaps (OCP violations) and accidental gaps (non-OCP violating, rare consonant combinations)?

- balanced variables: expected probability, neighborhood density, bigram probability
- independent variable: OCP violation or not
- result/conclusion: OCP had a significant effect on wordlikeness ratings, accounting for approximately 21% of subject variability; so subjects distinguish between systematic and accidental gaps

Experiment 3. Do subject acceptability judgments of OCP violations correlate with different degrees of featural similarity?

- balanced variables: expected probability, neighborhood density, and bigram probability
- independent variable: phonological similarity of homorganic consonants
- result/conclusion: similarity had a significant effect on wordlikeness rating (approximately 20% of subject variability); OCP is gradient

The Assessor module assigns acceptability scores to nonactual roots, so its assessment of nonsense words can be compared directly to the native speakers' judgments of nonsense words. To do this, we conducted the same tests from (Frisch and Zawaydeh, 2001), but substituted Assessor module acceptability ratings for their wordlikeness ratings. The hidden layer of the Assessor module can have any number of nodes, but we have investigated learning with hidden layers of between 1 and 10 hidden layer units and found that a range between 2 and 5 units produces effects parallel to the judgment data. Table 3 gives the results of a 5 unit hidden layer on three separate learning trials. All effects with significance at $p < .05$ are reported with the percentage of the variation accounted for by this effect. Under the Experiment 1 column, which used most of the stimuli, the correlation coefficients between Assessor outputs and Frisch and Zawaydeh's wordlikeness data (mean rating) is given as a gross measure of the correlation between the two datasets.

Table 3. Significant effects on acceptability from factors in Frisch & Zawaydeh 2001 experiments; cells show factor, percentage explained, and for experiment 1, correlation with the wordlikeness judgement data; network trained on Buckwalter 1997 corpus.

	Experiment 1, p<0.001	Experiment 2, p<0.001	Experiment 3, p<0.05
Trial 1	OCP 44%; 0.37	OCP 47%	similarity 5% (not sig.)
Trial 2	OCP 47%; 0.48	OCP 43%	similarity 9%
Trial 3	OCP 48%, 0.40	OCP 31%	similarity 17%

These results are qualitatively parallel to the Frisch and Zawaydeh's findings. In particular, in Frisch and Zawaydeh's experiment 1, OCP violation was the only statistically significant factor on wordlikeness and it still had a significant effect when bigram probability was controlled for in experiment 2. The results shown in Table 3 are therefore consistent with all these experimental findings, as OCP violation was the only significant factor in experiments 1 and 2, and similarity was a significant factor in two of the three trials of experiment 3. We note that the percentage of the acceptability explained by the OCP is slightly higher than with the judgement data in experiments 1 and 2, but we believe that a perfect match of the two datasets is not required to demonstrate induction of OCP constraints. A c-net model and learning protocol could be constructed to produce a better quantitative match with the experimental data through manipulation of model parameters and additional training, but we believe that such an effect would not be particularly revealing in this case. The important finding is therefore that a relatively simple set of parameters reproduces all of the statistically significant generalizations in the behavioral data.

5.2 Analysis of the hidden layer units

Up to this point, we have shown that our c-net learner can acquire a set of weights that allow it to approximate judgments of Arabic native speakers. At the symbolic level these judgments are described as knowledge of certain OCP-Place constraints (see section 2). Here, we illustrate that these constraints can be isolated as connectionist units in the hidden layer of the Assessor network. In doing so we demonstrate how the effects of symbolic constraints can be achieved through subsymbolic learning, i.e., the macro-structure constraints are psychologically real, but can be learned at the level of micro-structure.

When we know the architecture and weight matrices for a mature c-net, we can very easily compute the output for any input we choose, as we repeatedly do when training and testing the network. Interpreting at the symbolic level what this computation consists of is considerably more difficult, but nonetheless important to addressing the problem of constraint induction. One way to extract rule-like symbolic behavior from the rich dataset provided by our c-net is to use Classification and Regression Trees (CART). CART analysis is a commonly used statistical technique for imposing categorical structure on large and 'messy' datasets. For example, CARTs are used to make categorical medical decisions that are based on a patient's DNA make-up, a structure that is far rich for analysis by hand. CART analysis provides a categorical analysis by constructing a decision tree for this large dataset. In particular, it takes a dataset consisting of input variables and output variables and constructs a decision tree that attempts to predict the outputs from the inputs. The tree produced is not guaranteed to be optimal, but heuristics are used to obtain a tree that fits the data well (Hastie et al., 2009).

We applied the following method to produce CART trees for the five-node network described above. For each hidden node in the trained network, we applied the `classregtree` function of Matlab in order to produce decision trees that would predict the output from the input. The inputs for CART analysis were the 21,952 possible triliteral roots (i.e., both actual and nonactual roots). The specific variables for these inputs were just the 51 feature specifications used by the c-net to describe each triliteral root (i.e., distributed representations for 3 segments × 17 features). The output variables were the activation values for a particular hidden node, rounded to -1 or 1, a '1' meaning that the node judges the form favorably, '-1' unfavorably. This process produced five trees for the five hidden layer nodes.

The algorithm begins by identifying the single input variable, i.e., a phonological feature in a particular position, that does best at predicting the outputs. The data is then partitioned into two sets based on the value of this input feature. This procedure is then repeated recursively on each of the two sets, selecting a new input variable (=another feature in a C slot) to partition each set. We set the algorithm to stop when a set has fewer than 1,000 roots or else when all the output variables in a set are identical. At this point each terminal node is labeled either a '-1' or a '1', depending on which output variable predominates in the node. Once this is complete, the tree can be used to predict the output for a given input by descending the tree according to the input variables and then using the label for the resulting terminal node. For each hidden node, the CART tree did not have pure terminal nodes, meaning that the tree imperfectly predicted the output of the node over all the data. However, this coarse-graining effect on the function computed by the hidden layer nodes is helpful in interpreting the dominant trends in its behavior.

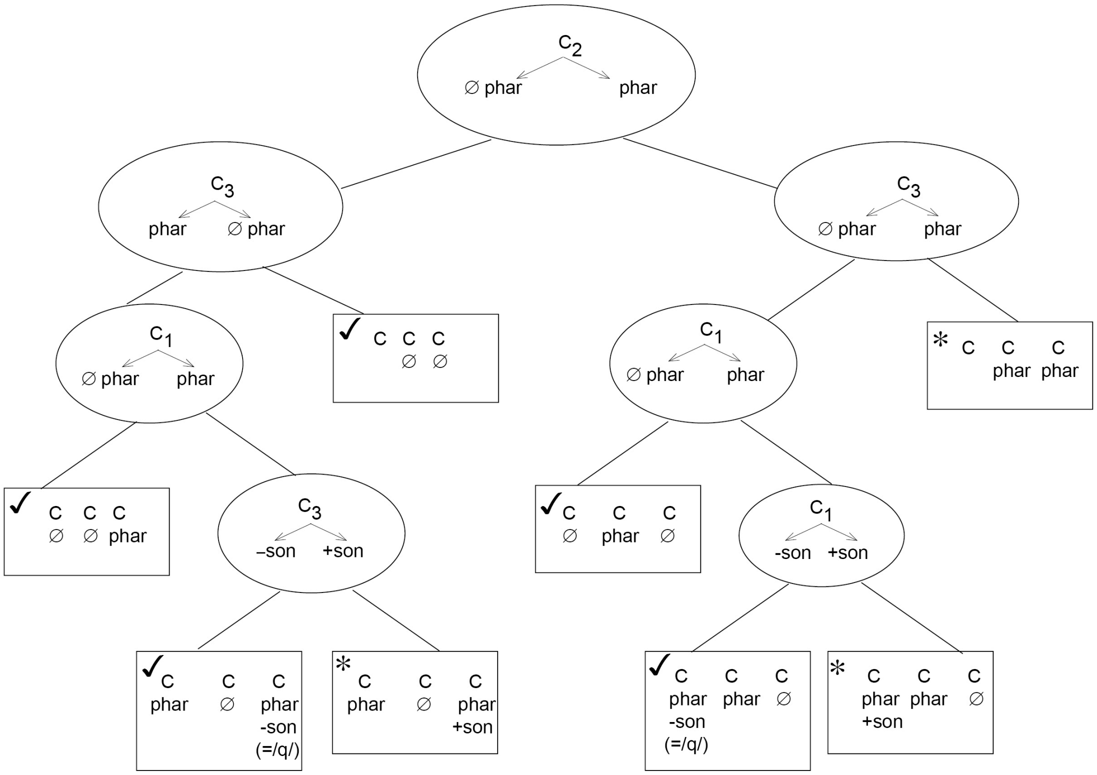

Figure 5. CART visualization of hidden layer node approximating OCP-pharyngeal. Circled nodes represent decisions about feature realization for a specified consonant, and boxed nodes represent predominant acceptable (checked) and marginal (starred) triliterals.

We applied this procedure of generating CARTs for all five hidden layer nodes, for three different trials. With one exception, each of the hidden nodes implements an approximation of one of the six OCP-Place coocurrence restrictions active in Arabic, i.e., OCP-labial, OCP-coronal/sonorant, OCP-coronal/fricative, OCP-dorsal, and OCP-pharyngeal (the latter two overlap with uvulars, as expected). In other words, in virtually all cases, the hidden layer nodes compute symbolic-like OCP constraints. As an example, Fig. 5 shows the CART tree for the fifth hidden layer node of the 1st trial. This node approximates the OCP for [pharyngeal] specification.

Two observations can be made about the CART visualization above for the OCP-pharyngeal node, and these observations are typical of the rest of the CARTs. First, this particular hidden layer node does a reasonably good job of predicting the nonoccurrence of triliterals that contain two pharyngeals. As we descend down the tree in Fig. 3, all triliterals (the boxed terminal nodes) that do not have two [phar] specifications (shown with at least two ∅) are allowed, and all but two of the schematic roots that have two [phar] are starred. These two apparent exceptions involve the segment /q/ in either C1 or C3. Recall that, to be consistent with prior work, all uvulars, including /q/, have a [pharyngeal] specification. These apparent exceptions reveal an important descriptive generalization in the dataset. While roots with two pharyngeals have low O/E values, most of the exceptions to OCP-pharyngeal, which again includes all uvulars, involve roots that contain /q/. There are 132 roots that contain /q/ and another pharyngeal consonant in the Buckwalter corpus, including 15 roots fitting the exempted pattern /q + Pharyngeal + C/ and 28 matching the pattern /Pharyngeal + C + q/. This fact is why /q/ is traditionally grouped with velars in descriptive statements of consonant cooccurrence (Greenberg, 1950). To summarize, the hidden nodes are capable of approximating the functions of symbolic constraints, even when segment-level exceptions exist. In the next section, we investigate the c-net's sensitivity to a host of segment-level exceptions that are far less robust statistically.

The CART trees above are useful for visualizing the coarse-grained nature of the functions computed by the hidden layer units. But since the CART only approximates the output of the c-net, it does not give an exact description of the behavior of a given node. To precisely quantify the relationship between the activations of hidden layer units and the OCP, the following method was used. First, we computed, for all possible triliteral roots, the violations of the specific OCP-place restrictions documented in Table 1. In particular, a violation of OCP-place requires adjacent segments of the same class in one of the same-place classes given below in Table 5. We assign a score of '-1' for that root and that class if there is an OCP violation, and a '1' otherwise. To correlate these facts with the behavior of the hidden layer, we computed the output of each hidden layer node times the weight of the connection leading out of it, for each root. This gives the acceptability of that root, as assessed by a specific hidden layer node. A positive acceptability indicates that the node thinks the word is well-formed; a negative value corresponds to ill-formedness. We then compute the correlation coefficient of each OCP value and each hidden node's acceptability. These correlations are shown for Trial 1 in Table 5 below (the same trial used for CART visualization in Figure 5). Results are qualitatively similar for the other trials. In Table 5 a value of '1' under a node column would indicate that a node does a perfect job of determing whether there is an OCP violation, and '-1' indicates the opposite. The final column shows the correlation between the activation of the output node and the various OCP violations. This last column gives an idea of how well the entire network approximates the OCP with respect to each place of articulation. All of the cells in the last column are positive (as they are for all three trials), showing that the network assigns on average lower acceptability scores for roots that violate these specific OCP constraints than otherwise.

Table 5. Correlations between OCP violation (differentiated by place class) and weighted activations of five hidden nodes and the output node. Results from Trial 1 with training on the Buckwalter corpus. Correlations greater than 0.1 are highlighted.

Class	Node 1	Node 2	Node 3	Node 4	Node 5	Output
OCP-labial	0.0829	-0.0850	0.0583	0.1138	-0.0645	0.0950
OCP-cor/stop	0.0635	0.0214	0.0616	0.1621	-0.0859	0.1455
OCP-cor/fric	-0.1272	0.4420	0.0945	0.2860	-0.1517	0.3356
OCP-dorsal	0.0245	-0.1277	0.0948	-0.0929	0.2544	0.0764
OCP-phar	0.1727	-0.1408	0.1160	-0.3814	0.7191	0.1798
OCP-cor/son	0.0852	-0.0555	-0.0062	0.1212	-0.0645	0.0744

For all rows, there is a positive correlation greater than .1 between a specific OCP violation and node activation (the bold-boxed cells). In some cases, a particular node stands out as doing the work of a specific OCP-place constraint, like node 5 for OCP-dorsal. In others, the effect of the OCP for a specific place class is spread over more than one node, for example, OCP-coronal/fricative. Furthermore, there are no strong negative correlations found in the other nodes that overpower these constraints. The highest negative correlation for the trial shown here is -0.3814 for node 4, a real outlier compared to the other negative values. But this is not strong enough to trump the effect of nodes 1, 3, and 5 in this network, which collectively approximate OCP-pharyngeal quite well.

5.3 Exceptional patterns

In section 2, we documented exceptions to the OCP in Arabic with a set of consonant specific patterns. The finding is that, for a pair of particular consonants, in a particular position, the distribution of the exceptions is not random. Rather, the exceptions tend to occur in specific bigram templates. For example, out of the 21 exceptions to the rule *Labial-C-Labial, 9 are /bCm/ and 11 are /fCm/. Only 1 is /bCf/, and there are none from /mCf/, /mCb/, or /fCb/. A natural question to ask, given the network's ability to extract out /q/ from OCP-Pharyngeal constraints, is whether our c-net is sensitive to these more fine-grained exceptional patterns.

We investigate this question by comparing the Assessor's score for the exceptional patterns we have identified in Table 2 (i.e., OCP-violating patterns that occur in the lexicon), with sets of roots that violate the OCP and that do not occur in Arabic. For example, to see if the network is sensitive to exceptions of the form /fCm/, versus other similar OCP[Lab] violations, we compute the acceptability scores for all 28 roots of the form /fCm/ (28 because there are 28 consonants that could replace C), and compare them with the scores for all roots of the form XCY where X and Y are non-identical labials, and excluding all /fCm/ forms. There are 5×28 of these nonexceptional roots, because there are five other logically possible combinations of the three non-identical labials. For each consonant pair, we take the average of the score for all roots with the exceptional pair, minus the average score for all roots that violate the OCP in the same way, but with a different consonant pair. For each consonant pair and each of one of three trails of the Assessor, we get a difference in means.

The results show that there was no clear pattern to the differences between the exceptional patterns and the logically possible but unattested patterns. Thus, in each trial approximately 30 out of 81 of the exceptional patterns in Table 2 were viewed less favorably than the comparable non-exceptional pattern, contrary to the hypothesis that the network might not view (attested) exceptional patterns as violators of the OCP. Averaging the difference over all exceptional pairs yielded a mean difference of acceptability score of approximately 0.05, which is quite small relative to the difference in score between OCP-violating and OCP-compliant roots. In sum, our c-net after training does not seem to be sensitive to the fine-grained exceptional patterns in Table 2 as a whole.

Just because a pattern exists in the lexicon, however, does not mean that it is part of a native speaker's phonotactic intuitions. For example, in the English lexicon there are no instances of diphthongs before palato-alveolar fricatives, but when speakers are tested for awareness of this constraint, (for example, by comparing *foushert* with *fousert*) there is no statistically significant difference in their rankings (Hayes, 2010). We cannot directly answer the question of whether native speakers of Arabic are sensitive to the patterns in Table 2 because the experiments in (Frisch and Zawaydeh, 2001) were not designed to answer this question. But the pilot data available from this study does not seem to provide any support for the contention that speakers have strong intuitions of the exceptional patterns. Thus, there were 19 nonsense roots in Frisch and Zawaydeh's study that fit the templates for exceptional patterns in Table 2, and the mean of these averaged wordlikeness scores is 2.6936. The mean of mean ratings of roots (*n*=64) that do not fit these patterns is slightly lower at 2.4591. This is consistent with the hypothesis that native speakers rate higher the roots that fit the attested exceptional patterns, but it is impossible to tell if this difference is meaningful, given the small number of examples and inability to pair the roots.

It is possible to group classes of roots, namely certain place classes that fit the exceptional patterns, and compare their means with the means of roots that fit nonexceptional patterns. Table 6 lists the mean ratings for three non-coronal place groups (there were too few examples of roots with coronal pairs) and shows the mean ratings for exceptional vs. nonexceptional roots and the difference of means. Except perhaps for dorsals, the data again does not show a significant trend.

Table 6. Mean wordlikeness judgments of roots with exceptional and non-exceptional OCP violating roots aggregated by select place classes.

	Labial	Dorsal	Pharyngeal	Totals
Exceptional	2.9583	2.9275	2.455	2.7803
Non-expectional	3.1403	1.9028	2.1034	2.3822
Differences	-0.182	1.0248	0.3216	

While it is possible that native speakers are sensitive to a subset of the exceptional patterns in Table 2, we believe that the lack of evidence for a trend in this pilot data supports a conjecture that native speakers are in fact not sensitive to many of the facts at this level of phonotactic detail. This is consistent with other findings, e.g., ((Hayes, 2010), (Becker et al., 2011)), and establishes a clear set of patterns that can be investigated in future experimental research. This conjecture is also consistent with our modeling results.

6. Discussion

This article has provided a cognitive architecture that makes learning the identity of grammatical constraints a significant part of learning. The web of connections in the Assessor module is a space of possible constraints, or a search space in the sense commonly used machine learning. Different configurations of connection weights constitute different subsymbolic constraint systems. When the correct configurations are learned, the larger network can be said to have learned the target constraint system. We have shown that a two layer feed-forward network can learn the phonotactic constraints of Arabic root phonotactics by properly setting the connection weights leading into and out of a set of hidden layer units. In other words, we have shown that the functions computed by these hidden layer units after training approximate quite well the functions computed by symbolic OCP-Place constraints familiar from generative phonology. The hidden layer units do not *exactly* compute OCP-Place constraints, but this finding is consistent with the data because Arabic cooccurrence restrictions are gradient in nature and have many exceptions. The larger finding is thus that the identity of phonotactic constraints themselves can be learned from data in this case, and do not have to be stipulated in advance.

This result sets our connectionist learning system apart from many contemporary approaches to learning phonotactics. As summarized above, most constraint-ranking algorithms can find the correct ranking of constraints, given the right data and a reasonable amount of time ((Tesar, 2004), (Prince and Tesar, 2004); (Boersma, 1998), (Boersma and Hayes, 2001); (Pater, 2009)). But these investigations do not make learning the constraints themselves part of the learning problem. Furthermore, it has been conjectured that there is a close parallelism between the macro-structure of OT grammars and the micro-structure of connectionist networks (Smolensky and Legendre, 2006b). However, as stated at the outset of this important work (chapter 1, section 2), the connectionist implementations of OT constraint systems have not yet shown how behavior resembling symbolic constraint interaction can be learned at this level of explanation. Our contribution to Smolensky and Legendre's research paradigm is thus to show that at least one kind of phonological pattern, place-based cooccurrence restrictions, can be learned at the micro-structure level.

The finding that constraints can be learned from data supports a comparison of our model to the MaxEnt phonotactic learning paradigm. Both approaches use principles of statistical learning to search a vast constraint space and provide the constraints that give a good approximation of the target grammar. As such, both rely heavily on a suitably large and representative data sample. Another similarity is that both approaches produce inductive baselines, or simple systems derived from data. These systems have limits, for example, the generalizations involving suprasegmentals that Hayes and Wilson document in their study. We have explored select problems in Arabic consonant phonology that extend the core system we document here, and have found that there are also certain limits to our simple two layer system. For example, it is a basic fact of Arabic that partially similar segments are avoided, but identical segments in C2C3 position slip by the OCP constraints. This is a kind of nonlinear function that a multilayer c-net ought to be able to learn and compute. Compare its non linear separability (e.g., *p-b, *b-p vs p-p, b-b) with that of exclusive 'or' (0-1, 1-0 vs 0-0, 1-1). Yet even after extensive parameter switching with our Assessor module, pairs of identical segments are not assessed correctly if we treat them as triliterals. These simulations support the conclusions of (Berent and Shimron, 1997), based on similar data from Hebrew, that linguistic constituency is necessary to express the generalization about final geminates. Similarly, the segment pair templates documented in section 2 seem to be too fine-grained for our basic system to learn, though at present we do not know if native speakers also have intuitions of these templates.

We do not take these findings as insurmountable obstacles for connectionist learning models. Rather, like Hayes & Wilson, we believe they motivate additional theoretical assumptions and structure in our model. Indeed, our network is about as simple as it can get, short of a one-layer perceptron, and could be extended in a host of ways. For example, inputs could be endowed with constituency (syllables, feature classes, etc.) with tensor product representations ((Smolensky, 1990), (Smolensky and Legendre, 2006a)) to allow for generalization based on the role of an element in a constituent. Other extensions of our basic system are inclusion of a context layer in a sequential network architecture (as employed by (Hare, 1990)), recurrent connections for modeling problems that are time-based and therefore require more detailed dynamics, and simply including a larger number of hidden layers.

How does our c-net model differ from the MaxEnt approach generally? A basic distinction can be made by pointing out that the MaxEnt approach uses symbolic constraints and c-nets use subsymbolic constraints. In theory, this difference has empirical consequences, because the c-net constraint space is uncountably infinite and so it is richer. We actually do not believe that this formal difference has empirical consequences that matter for the study of language. The difference between a constraint space of e.g., 300,000 symbolic constraints and an infinite set of subsymbolic constraints is not likely to matter for most problems in generative linguistics once constraint weights are assigned to the constraints selected in a MaxEnt grammar. One might also remark that c-net learning is inherently gradual in that adjustments are made to the whole network after processing each form, while MaxEnt learning, at least as it is implemented, involves processing whole sets of language forms collectively. We also do not think this is a theoretical difference of much significance, as there is nothing in principle that prevents an on-line version of MaxEnt learning. Indeed, Colin Wilson (personal communication) informs that such an algorithm exists.

We think that one aspect of our approach that sets it apart from other paradigms, however, is the potential for integration with psycholinguistic models of production and perception. In the spreading interactive model of (Dell, 1986), for example, selection of a word in the mental lexicon is simulated as the spreading of activation through a lexical network of many linguistic layers (i.e., morphological and phonological constituents). While there are important features of this model that differ from our c-net, e.g., bidirectional spreading and rich linguistic representations, an important point is that lexical selection is the result of activation spreading, an output pattern predicted by parallel processing of micro-elements. Another important model is the TRACE theory of word recognition (McClelland and Elman, 1986), which uses spreading activation and parallel distributed processing to work in the other direction, predicting word forms from phonetic attributes. These models have been tremendously influential and provided a set of assumptions shared with many contemporary theories of speech production and perception. We believe that the parallel-distributed processing principles at the core of these two influential theories and our c-net may allow for a more natural integration of the functions of our c-net within these models. Furthermore, this integration is highly desirable in the case of dissimilatory phenomena, like Arabic OCP-Place constraints. As shown in detail in (Frisch, 1996), (Frisch et al., 2004), (Frisch, 2004), and (Martin, 2007), many of the properties of dissimilatory patterns can be explained as the long term diachronic effects of constraints on speech production and perception.

A step in this direction was made in (Frisch, 1996) by showing a close mathematical relationship between his natural class similarity model of English and Arabic and the similarity structure predicted by spreading activation in a Dell-style production system. Indeed, the natural class model is used in this work as a closed form approximation of a connectionist network, with the hope that future research will develop mathematically accessible connectionist models of native speaker intuitions. The present work makes a further step by showing how grammatical constraints can be represented and learned in a two-layer feed-forward network. However, problems not specific to connectionism preclude a realization of Frisch's original vision. We have already mentioned in section 3 the problem of separating production and assessment of phonotactic intuitions. The consequence of our decision to separate the two modules is that our Assessor module produces acceptability scores, not linguistic representations, like the two foundational theories above. These psycholinguistic models therefore compute different functions, which must be addressed to bring our c-net closer to these models.

Another central problem is the encoding of serial order and the formalization of competitive inhibition in language processing. Our model is non-sequential in the sense that it feeds the Autoassociator and Assessor modules a preordered string of consonants. Our network is similar to the spreading activation and TRACE models, but it distinguishes it from other connectionist networks that model serial order as sequential output conditioned by a context layer ((Jordan, 1991), (Elman, 1990)); see also (Dell et al., 1997). Again, our network is non-sequential because we simply do not address the serial order problem here, but several important psycholinguistic effects depend on a competition between nodes that results with a formalization of sequences (Frisch, 2004). We hope that future research can relate our findings on constraint induction to the proper characterization of serial order in respective domains of language processing.

References

Becker, Michael, Ketrez, Nihan, and Nevins, Andrew. 2011. The surfeit of the stimulus: Analytic biases filter lexical statistics in Turkish laryngeal alternations. *Language* 87:84-125.

Berent, Iris, and Shimron, Joseph. 1997. The representation of Hebrew words: Evidence from the obligatory contour principle. *Cognition* 64:39-72.

Boersma, Paul. 1998. *Functional Phonology*. The Hague: Holland Academic Graphics.

Boersma, Paul, and Hayes, Bruce. 2001. Empirical tests of the Gradual Learning Algorithm. *Linguistic Inquiry* 32:45-86.

Bowers, J. S. 2009. On the biological plausibility of grandmother cells: Implications for neural network theories in psychology and neuroscience. *Psychological Review* 116:220-251.

Buckwalter, Tim. 1997. The triliteral and quadriliteral roots of Arabic. URL: http://www.angelfire.com/tx4/lisan/roots1.htm.

Bybee, Joan, and McClelland, James L. 2005. Alternatives to the combinatorial paradigm of linguistic theory based on domain general principles of human cognition. *The Linguistic Review* 22:381-410.

Clements, G.N., and Hume, Elizabeth V. 1995. The internal organization of speech sounds. In *The handbook of phonological theory*, ed. John A. Goldsmith, 245-306. Cambridge, MA: Blackwell.

Coetzee, Andries, and Pater, Joe. 2008. Weighted constraints and gradient restrictions on place co-ccurrence in Muna and Arabic. *Natural Language and Linguistic Theory* 84:289-337.

Cowan, J. Milton (ed.). 1979. *Hans Wehr: A dictionary of Modern Written Arabic*. Wiesbaden, Germany: Otto Harrasowitz.

Dell, Gary S. 1986. A spreading interactive theory of retrieval in sentence production. *Psychological Review* 93:283-321.

Dell, Gary S., Burger, L. K., and Svec, W. R. 1997. Language production and serial order: A functional analysis and a model. *Psychological Review* 104:123-147.

Elman, Jeffrey. 1990. Finding structure in time. *Cognitive Science* 14:179-211.

Elman, Jeffrey, Bates, Elizabeth, Johnson, Mark, Karmiloff-Smith, Annette, Parisi, Domenico, and Plunkett, Kim. 1996. *Rethinking innateness: A connectionist perspective on development*. Cambridge MA: MIT Press.

Frisch, Stefan. 1996. *Similarity and frequency in phonology*: Ph.D. dissertation, Northwestern University, Evanston, IL.

Frisch, Stefan, and Zawaydeh, Bushra. 2001. The psychological reality of OCP-Place in Arabic. *Language* 77: 91-106.

Frisch, Stefan A. 2004. Language processing and segmental OCP effects. In *Phonetically-based phonology*, eds. Bruce Hayes, Robert Kirchner and Donca Steriade, 346-371. Cambridge: Cambridge University Press.

Frisch, Stefan A., Large, Nathan R, and Pisoni, David S. 2000. Perception of wordlikeness: Effects of segment probability and length on the processing of nonwords. *Journal of Memory and Language* 42:481-496.

Frisch, Stefan A., Pierrehumbert, Janet, and Broe, Michael B. 2004. Similarity avoidance and the OCP. *Natural Language and Linguistic Theory* 22:179-228.

Gafos, Adamantios. 1998. Eliminating long-distance consonantal spreading. *Natural language and linguistic theory* 16.2:223-278.

Goldrick, Matthew, and Daland, Robert. 2009. Linking speech errors and phonological grammars: Insights from Harmonic Grammar networks. *Phonology* 26:147-185.

Goldsmith, John. 1976. Autosegmental phonology, MIT: Doctoral dissertation.

Gomez, Rebecca L. 2002. Variability and detection of invariant structure. *Psychological Science* 13:413-436.

Greenberg, Joseph. 1950. The patterning of root morphemes in Semitic. *Word* 6:162-181.

Hare, Mary. 1990. The role of similarity in Hungarian vowel harmony: A connectionist account. In *Connectionist natural language processing*, ed. Noel Sharkey, 295-322. Oxford: Intellect.

Hastie, Trevor, Tibshirani, Robert, and Friedman, Jerome. 2009. *The elements of statistical learning*. New York: Springer.

Hayes, Bruce, and Wilson, Colin. 2008. A maximum entropy model of phonotactics and phonotactic learning. *Linguistic Inquiry* 39:379-440.

Hayes, Bruce. 2010. Learning-theoretic linguistics: Some examples from phonology. Presentation given at 2010 Cognitive Science Conference, session in honor of Rumelhart Prize winner James McClelland. Ms.

Jordan, Michael I. 1991. Serial order: A parallel distributed processing approach. In *Advances in connectionist theory: Speech*, eds. Jeffrey Elman and David Rumelhard, 214-249. Hillsdale, NJ: Erlbaum.

Leben, Will. 1973. Suprasegmental Phonology, MIT: Doctoral dissertation.

Legendre, Géraldine, Miyata, Yoshiro, and Smolensky, Paul. 1990. Can connectionism contribute to syntax? Harmonic Grammar, with an application. In *Proceedings of the 26th Regional Meeting of the Chicago Linguistic Society*, eds. M Ziolkowski, M Noske and K Deaton, 237-252. Chicago: Chicago Linguistic Society.

Legendre, Géraldine, Sorace, Antonella, and Smolensky, Paul. 2006. The Optimality Theory-Harmonic Grammar connection. In *The harmonic mind: From neural computation to Optimality Theoretic grammar*, eds. Paul Smolensky and Géraldine Legendre, 339-402. Cambridge, MA: The MIT Press.

Martin, Andy. 2007. The evolving lexicon, University of California, Los Angeles.

McCarthy, John J. 1979. Formal problems in Semitic phonology and morphology, MIT: Doctoral dissertation.

McCarthy, John J. 1986. OCP Effects: Gemination and antigemination. *Linguistic Inquiry* 17:207-263.

McCarthy, John J. 1988. Feature geometry and dependency: A review. *Phonetica* 43:84-108.

McCarthy, John J., and Prince, Alan. 1990. Prosodic morphology and templatic morphology. In *Perspectives on Arabic linguistics II: Papers from the Second Annual Symposium on Arabic Linguistics*, eds. M. Eid and John McCarthy, 1-54. Amsterdam: John Benjamins.

McCarthy, John J. 1994. The phonetics and phonology of Semitic pharyngeals. In *Papers in Laboratory Phonology III*, ed. Patricia A. Keating, 191-233. Cambridge: Cambridge University Press.

McCarthy, John J., and Prince, Alan. 1999. Faithfulness and identity in Prosodic Morphology. In *The prosody-morphology interface*, eds. René Kager, Harry van der Hulst and Wim Zonneveld, 218-309. Cambridge: Cambridge University Press.

McClelland, James L., and Elman, Jeffrey. 1986. The TRACE model of speech perception. *Cognitive Psychology* 18:1-86.

McClelland, James L., and Rumelhart, David E. 1986. On learning the past tenses of English verbs. In *Parallel Distributed Processing: Explorations in the microstructure of cognition, Volume 2: Psychological and biological models*, eds. James L. McClelland, David E. Rumelhart and The PDP Research Group, 216-271 Cambridge, MA: The MIT Press.

McLeod, Peter, Plunkett, Kim, and Rolls, Edmund T. 1998. *Introduction to connectionist modelling of cognitive processes*. Oxford: Oxford University Press.

Mitchell, Tom M. 1997. *Machine learning*. Boston, MA: McGaw Hill.

Myers, Scott. 1997. OCP effects in Optimality Theory. *Natural Language and Linguistic Theory* 15:847-892.

Newport, Elissa, and Aslin, Richard. 2004. Learning at a distance: I. Statistical learning of non-adjacent dependencies. *Cognitive Psychology* 48:127-162.

Padgett, Jaye. 1995. *Stricture in feature geometry*: Dissertations in linguistics. Stanford, Calif.: CSLI Publications.

Pater, Joe. 2009. Weighted constraints in generative linguistics. *Cognitive Science* 33:999-1035.

Pierrehumbert, Janet. 1993. Dissimilarity in the Arabic verbal roots. In *NELS 23*, 367-381.

Prince, Alan, and Smolensky, Paul. 1993/2004. *Optimality theory: Constraint interaction in generative grammar*. Malden, MA: Blackwell.

Prince, Alan, and Tesar, Bruce. 2004. Learning phonotactic distributions. In *Fixing priorities: Constraints in phonological acquisition*, eds. René Kager and Joe Pater, 245-291. Cambridge: Cambridge University Press.

Ramsey, William, Stich, Stephen, and Garon, Joseph. 1990. Connectionism, eliminativism and the future of folk psychology. In *Connectionism: Debates on folk psychology*, eds. C. Macdonald and G. Macdonald, 311-338. Cambridge, MA: Basil Blackwell.

Rose, Sharon. 2000. Rethinking geminates, long-distance geminates and the OCP. *Linguistic Inquiry* 31:85-122.

Rumelhart, David, Hinton, Geoffrey E, and Williams, Ronald J. 1986a. Learning internal representations by error propagation. In *Parallel distributed processing: Explorations in the microstructure of cognition. Vol 1-2*, eds. James L. McClelland, David Rumelhard and The PDP Group, 318-362. Cambridge: The MIT Press.

Rumelhart, David, McClelland, James L., and Group, The PDP Research. 1986b. *Parallel distributed processing: Explorations in the microstructure of cognition. Volumes 1-2.* . Cambridge, MA: MIT Press.

Ryding, Karin C. 2005. *A reference grammar of Modern Standard Arabic*. Cambridge: Cambridge University Press.

Smolensky, Paul. 1988. On the proper treatment of connectionism. *The Brain and Behavioral Sciences* 11:1-23.

Smolensky, Paul. 1990. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence* 46:156-216.

Smolensky, Paul, and Legendre, Géraldine. 2006a. Formalizing the principles II: Optimization and grammar. In *The harmonic mind, From neural computation to optimality-theoretic grammar. Vol 1: Cognitive architecture*, eds. Paul Smolensky and Géraldine Legendre. Cambridge, MA: The MIT Press.

Smolensky, Paul, and Legendre, Géraldine. 2006b. *The harmonic mind. From neural computation to optimality theoretic grammar*. Cambridge, MA: The MIT Press.

Smolensky, Paul, Goldrick, Matthew, and Mathis, Donald. To appear. Optimization and quantization in gradient symbol systems: A framework for integrating the continuous and the discrete in cognition. *Cognitive Science*.

Spencer, John P., Samuelson, Larissa K., Blumberg, Mark S., McMurray, Bob, Robinson, Scott R., and Tomblin, Bruce J. 2009. Seeing the world through a third eye: Developmental Systems Theory looks beyond the nativist-empiricist debate. *Child Development Perspectives* 3:103-105.

Suzuki, Keiichiro. 1998. A Typological Investigation of Dissimilation, University of Arizona: Doctoral dissertation.

Tesar, Bruce. 1995. Computational Optimality Theory, University of Colorado: Doctoral dissertation.

Tesar, Bruce, and Smolensky, Paul. 2000. *Learnability in Optimality Theory*. Cambridge, MA: MIT Press.

Tesar, Bruce. 2004. Using inconsistency detection to overcome structural ambiguity in language learning. *Linguistic Inquiry* 35:219-253.

Thomas, Michael S. C., and McClelland, James L. 2008. Connectionist models of cognition. In *Cambridge handbook of computational psychology*, ed. Ron Sun, 23-58. Cambridge: Cambridge University Press.

Wayment, Adam. 2009. Assimilation as attraction: Computing distance, similarity, and locality in phonology, Johns Hopkins University.

Yip, Moira. 1989. Feature geometry and cooccurrence restrictions. *Phonology* 6:349-374.

Zamuner, Tania, Kerkhoff, Annemarie, and Fikkert, Paula. 2006. Phonotactic and morphological acquisition. Ms.