Connectionist Symbol Processing: Dead or Alive?

Contributors

D. S. Blank,	M.S. Cohen,	M. Coltheart,	J. Diederich,	B.M. Garner,	R.W. Gayler,
C.L. Giles,	L. Goldfarb,	M. Hadeishi,	B. Hazlehurst,	M. J. Healy,	J. Henderson,
N. G. Jani,	D. S. Levine,	S. Lucas,	T. Plate,	G. Reeke,	D. Roth,
L. Shastri,	J. Sougne,	R. Sun,	W. Tabor,	B. B. Thompson,	S. Wermter

Editors Arun Jagota, Tony Plate, Lokendra Shastri, Ron Sun

Producers Arun Jagota, Nigel Duffy

Preface

In August 1998 Dave Touretzky asked on the connectionists e-mailing list, "Is connectionist symbol processing dead?" This query lead to an interesting discussion and exchange of ideas. We thought it might be useful to capture this exchange in an article. We solicited contributions, and this collective article is the result.

Contributions were solicited by a public call on the connectionists e-mailing list. All contributions received were subjected to two to three informal reviews. Almost all were accepted with varying degrees of revision. Given the number and variety of contributions, the articles cover a wide, though by no means complete, range of the work in the field.

The pieces in this article are of varying nature: position summaries, individual research summaries, historical accounts, discussion of controversial issues, etc. We have not attempted to connect the various pieces together, or to organize them within a coherent framework. Despite this, we think, the reader will find this collection useful.

The Radical Alternative to Hybrid Systems

Douglas S. Blank

University of Arkansas, Fayetteville, USA, dblank@comp.uark.edu

Implementing symbol processing in networks was a good first step in solving many problems that plagued symbolic systems. Tony Plate's HRR as applied to analogy is a great example [147]. Using connectionist representations and methodologies, an expensive symbolic similarity estimation process was eliminated in the analogy-making MAC/FAC system [60].

Unfortunately, the entire MAC/FAC hybrid model (like many such models) has a fatal flaw that prevents it from leading to an autonomous, flexible, creative, intelligent (analogy-making) machine: the overall system organization is still rigidly "symbolic". Their method requires that analogies be encoded as symbols and structures, which leaves no room for perception or context effects during the analogy making process (for a detailed description of this problem, see Hofstadter [86]). For these reasons, implementing Gentner's rigid framework completely in a neural network (or even real neurons) won't help.

Plate's hybrid solution, like most hybrid systems, solved many problems of the MAC/FAC purely-symbolic system. No doubt, hybrid systems are better than their symbolic relatives. However, wherever symbols and structures remain, we seem to be faced with other problems of brittleness and rigidity.

[0] Neural Computing Surveys 2, 1-40, 1999, http ://www.icsi.berkeley.edu/~jagota/NCS

I believe that in order to tackle the big unsolved AI and cognitive science problems (like making analogies), we, as modelers, will have to face a radical idea: we will no longer understand how our models solve a problem exactly. I mean that, for many complex problems, systems that solve them won't be able to be broken down into symbols and modules, and, therefore, there may not be a description of the solution more abstract than the actual solution itself.

This is a radical view that many researchers would argue against. Istvan S. N. Berkeley sums up this opposition in the Connectionists mailing list:

> "It seems to me that there is something fundamentally wrong about the proposal here. As McCloskey [130] has argued, unless we can develop an understanding of how network models (or any kind of model for that matter) go about solving problems, they will not have any useful impact upon cognitive theorizing. Whilst this may not be a problem for those who wish to use networks merely as a technology, it surely must be a concern to those who wish to deploy networks in the furtherment of cognitive science. If we follow [Blank's suggestion] then even successful attempts at modelling will be theoretically sterile, as we will be creating nothing more than 'black boxes'."

It is true: if we follow this radical view, we will end up creating "black boxes" or, more radically, the "Big Black Box" solution. However, it is not true that we will be left with a "theoretically sterile" science of cognition. Instead of "theories" that explain "how it works" at a fine level, we will have explanations of "how it developed" at a coarse level. Such a theory would look very different from, say, Marr's theory of vision [127].

Many researchers have been focusing on such explanations by attempting to solve high-level problems via a purely connectionist framework. Some high-level systems that come to mind include Elman's developmental models, Meeden's planning system, and my own connectionist analogy-making system [43, 133, 20]. Rather than focusing on some assumed-necessary symbolically-based process (say, variable binding) these models look at a bigger goal: modeling a complex behavior.

My analogical model does not assume that analogies are correspondences made via searching through symbolic structures. Rather, a network is trained to recognize similar parts of images or sets of relations. The goal of the project was to see how far we can currently model a seemingly symbolic high-level behavior without resorting to the traditional symbolic assumptions.

Of course, many systems cannot be "black boxes". Military systems, or modules that interface with other subsystems may need to harness the explanative power and structure of symbolic systems. But the general cognitive scientist seeking the most intelligent, autonomous artificial agents possible need not be constrained in such a way. For us, there is the radical alternative.

Therefore, building and manipulating structured representations or binding variables via networks should not be our goal. Neither should creating a model such that we can understand its inner workings. Rather, we should focus on the techniques that allow a system to self-organize such that it can solve the bigger problems. Much of the research on "learning to learn" is, I believe, related to this issue.

Connectionist symbol processing has no doubt created more robust symbolic systems; connectionism has shown that it can be used as an engineering tool to create more flexible, less brittle symbolic processing. However, for me, "connectionist symbol processing" was just a very useful stage I went through as a cognitive scientist. Now I see that networks can do the equivalent of processing symbols, and not have anything to do with symbols. In addition, I learned that I can let go of the notion that I will understand exactly how a network does it.

Connectionism, Commerce and Cognition

Max Coltheart

Macquarie University, Sydney, Australia, max@currawong.bhs.mq.edu.au

There are at least four distinct spheres of endeavour in which one could seek evidence of successes in connectionist symbol processing. Because great success in any one of these spheres by no means guarantees any success at all in the others, we need to examine each separately.

The first of these spheres is the commercial. Connectionist networks have been applied to many problems of commercial interest - handwriting recognition, voice recognition, predicting the stock market - and it would be of interest to survey such applications to assess the degree to which they have been commercially successful.

The second domain is the formal theory of neural networks, which I take to be a branch of mathematics. This is concerned with proving theorems dealing with such matters as the particular types of symbol-processing problems which particular learning algorithms can or cannot learn. I don't think anyone would dispute the claim that we know far more about such matters now than we did twenty years ago.

Neither of these two domains of applicability of connectionism has anything to do with symbol-processing as it is carried out by living organisms, of course; however, the other two domains I discuss are concerned with this.

The third domain is the brain. Connectionist networks are often referred to as neural networks of course, and much connectionist research results in statements about brain function. What have we learned about how the brain processes symbols by doing connectionism instead of, for example, doing single-cell recording or psychopharmacology? This is far from clear, in my opinion, because much work of this kind argues in the following way: we know that the brain is a neural network, and the formal theory of neural networks tells us a great deal about what neural nets can and cannot do, and how they do what they do; therefore connectionism tells us a great deal about what the brain can and cannot do, and how it does what it does. Of course, the word "neural" in "the brain is a neural network" and the word "neural" in "the formal theory of neural networks" refer to quite different things. Nevertheless, one comes across arguments that do seem to take this form - for example, arguments to the effect that since formal neural networks are susceptible to catastrophic interference, the brain needs some way of avoiding such this problem, and that means that connectionism tells us what the hippocampus is for. This doesn't follow, because it doesn't follow that catastrophic interference must be a problem for the brain if it is a problem for formal neural networks.

Which brings us to the final domain, the mind - the functional or information-processing level of explanation of symbol processing.A major contribution that springs to mind here is the Interactive Activation Model (IAM) of letter and word recognition [129]. It offered an excellent quantitative account of a body of data from studies of word and letter perception; and it remains influential, since two contemporary computational models of word recognition [35, 67] are directly derived from the IAM model. Why, then, in the extensive discussion of connectionist symbol processing did no one offer the IAM as an example of a success in this field? This can't be because the IAM doesn't have anything to do with symbol processing, since letters and words are, obviously, symbols. So is the IAM not regarded as a connectionist model? It is hard to see why it doesn't qualify here, since the model consists of three layers of units which communicate with each other via excitatory and inhibitory connections. However,I suggest that this kind of model is regarded as nonconnectionist because of two of its properties. The first is that it uses local rather than distributed representation. The second is that its connection strengths are hardwired by the modellers, not learned via some learning algorithm. If I am right about this, then I conclude that our debate about connectionism and symbol processing has really been a debate about whether symbol processing can be done well by systems that use distributed representations and by systems that are trained by one of the usual learning algorithms (which sounds like two debates rather than one).

Connectionist Symbol Processing: Is There an Efficient Mapping of Feedforward Networks To a Set of Rules?

Joachim Diederich

Queensland University of Technology, Australia, joachim@moon.fit.qut.edu.au

Dave Touretzky asked about the current state of connectionist symbol processing in light of the fact "that we did not have good techniques for dealing with structured information in distributed form, or for doing tasks that require variable binding." Consequently, a number of contributors (e.g. Feldman, Shastri, Plate and others) have focused on rule-based or symbolic processing in neural networks. Rule-extraction from trained neural networks is a side-issue in this context. The objective is to efficiently map a neural network to a set of (minimal rules); the neural network as such is not used for rule-based processing. However, if it is possible to map a feedforward NN to a rule-set in polynomial time, then this is relevant for the current debate. For instance, it would shed new light on discussions such as the learning of the past-tense of English verbs and to what degree rule-based behavior emerges from feedforward networks. Since the underlying processing mechanism cannot be observed but must be inferred from empirical data for most cognitive tasks, knowledge of an efficient mapping of a feedforward network to a set of rules that closely approximate the behavior of the network is relevant. It not only touches the question of the appropriate level of abstraction for cognitive modeling but it makes certain questions (Is the underlying mechanism NN or rule-based?) almost impossible to answer. If the brain or any other computational system can efficiently map one representation (a neural network) into another (a set of rules), then it is close to impossible to infer the correct representation from empirical data.

Over the last years, the principal capabilities and limitations of the most important rule-extraction from NN techniques have been studied analytically (with respect to the computational complexity) and experimentally. Furthermore, rule-extraction from NN techniques have been compared with inductive inference methods (again, theoretically and by experiment). Noisy and incomplete data sets have been used for benchmarks in order to determine the suitability of these algorithms for real-world applications. The following is a brief summary of some relevant results:

Theory

Golea [66] showed that extracting the minimum DNF expression from a trained feedforward net is hard in the worst case. Furthermore, Golea [66] showed that the Craven & Shavlik [37] algorithm is not polynomial in the worst case. A promising approach is extracting the best rule, within a given class of rules, from single perceptrons. However, extracting the best N-of-M rule from a single-layer network is again hard. Maire [123] presents a method to unify rules extracted by use of the N-of-M technique. Independently, he shows that deciding whether a perceptron is symmetric with respect to two variables is NP-complete.

More recently Maire [124] introduced an algorithm for inverting the function realised by a feedforward network based on the inversion of the vector-valued functions computed by each individual layer. This new rule-extraction algorithm backpropagates regions from the output to the input layer. These regions are finite unions of polyhedra; applied over the input space they approximate to a user- defined level of accuracy the true reciprocal image of constraints on the outputs (similar to Validity Interval Analysis). A core problem of the rule-extraction task is thereby solved. The method can be applied to all feedforward networks independent of input-output representations.

Empirical studies

A number of rule-extraction from trained neural networks techniques have been compared by use of the standard benchmark data sets. In addition, rule-extraction from neural network techniques have been compared with symbolic machine learning methods. Good results have been reported for decompositional technique wrt. rule-quality. Results are generally poorer for pedagogical, learning-based techniques [8].

Symbols, Symbolic Processing and Neural Networks

B. M. Garner

RMIT, Melbourne, Australia, bmg@cs.rmit.edu.au

This contribution poses questions in an attempt to clarify the issues involved in symbolic processing and hypotheses are made as to the answers of these questions. These questions being, what is "symbol", what is "symbolic processing", and where does symbolic processing occur within the brain? Currently available training algorithms will train neural networks as sets of constraints and it is seen that this is consistent with the definition of symbol proposed here. The proposed definition of symbol is 'symbol is something that takes its meaning from those symbols "around" it'. Also when the right preprocessing on input data is done, even very complex problems can be learnt easily.

The Definition of symbol

When looking at symbols they are always defined in terms of relationships to other symbols, and that symbols do not occur by themselves. Often symbols are imbedded into other symbols, and can be used and reused and take their meaning only within the context they are found in. So perhaps the best definition of symbol is that 'symbol is something that takes its meaning from those symbols "around" it'. Perhaps there are better definitions because this one is self-referential. However, this idea of symbol is close to that proposed by structural linguistics (de Saussure [38], Leach [112], Levi-Strauss [113]).

While there may be some debate over a connectionist definition of what a symbol is, there needs to be some definition. Finding an adequate definition may not be easy as there is much debate from many disciplines, so it may be best to propose something and see how useful it is.

Symbolic Processing

Symbolic processing is where the symbols are either learnt so that they can be recalled at another time, or classified as having previously been learnt. This is referred to as learning/classification process in this contribution.

Two training algorithms for feedforward neural networks were published by Garner [56, 57], where the network was said to be trained to a symbolic solution because numeric values for the weights and the thresholds are not found. The network is trained into sets of constraints for each 'neuron' in the hidden and output layers. The constraints show that the weights and the thresholds are in relationship to each other at each 'neuron' [55]. This relationship is comparative, hence each weight or threshold takes its meaning from the weights in the neuron is belongs to. The proposed definition of symbol supports the results of available training algorithms [56, 57].

The constraints can be seen to define logic relationships within the neuron, then together with the other neurons in the trained network, form larger logic relationships. So after the network has been trained to learn a problem, it is possible to see how the network is structured to learn the symbols that the network is being trained to recognise.

Symbolic Input and Output

Before symbolic processing can take place within the network of neurons, the input has to be transformed into a form suitable for processing.

Input is presented to the brain in the form of sense data via the nervous system. This input can be viewed as a symbol. What is meant by this, is that the sensation of that which is being touched, for instance, is encoded and presented to the brain via nerve endings, and that sensation is the symbol, as it represents the object being touched. A similar process is occurring for all the senses the symbol encoding transforms the sensation to a format suitable for the brain to be classified or learnt; another example is the visual alphanumeric symbol 'a' which is distinct from the audible alphanumeric symbol 'a' have to somehow be translated into the same abstract symbol. Hence, the input is encoded and transformed into an abstract format that allows the information to be learnt and compared with other symbols.

This problem of transforming data to a format suitable to be learnt by an ANN is familiar to people who have trained a number of them. Even difficult problems such as the twin spiral problem can be learnt with one hidden layer if the input space is very well understood and abstracted appropriately for learning.

It would seem that some similar transformation, as in symbol input, takes place after classification/learning or symbolic processing for the purposes of thought, communication, action, and so on.

Conclusions

This has been an attempt clarifying the issues involved in symbolic processing, if connectionism is to seriously address this problem. It has been proposed that there are several mechanisms occurring, a number of very good transformation processes, and probably one very general learning process that will learn practically anything in a binary format.

Holographic Networks are Hiking the Foothills of Analogy

Ross W. Gayler

University of Melbourne, Australia, r.gayler@psych.unimelb.edu.au

Dave Touretzky asked whether connectionist symbol processing is dead. In expanding on this question he made the following observations: (i) "The problems of structured representations and variable binding have remained unsolved.", (ii) "The last really significant work in the area was Tony Plate's holographic reduced representations", and (iii) "No one is trying to build distributed connectionist reasoning systems any more, like the connectionist production system I built."

With respect to the first observation, I am not sure which problems Dave is referring to, but the fundamental problem of whether structured representations and variable binding are possible in a connectionist system was answered in the affirmative by Paul Smolensky [179]. His tensor product binding method allows arbitrary vectors to be bound to other vectors, implementing variable/value bindings and recursively composed structures. The method deals with fully distributed representations of bindings and handles local representations as special cases.

If the vectors being composed are chosen randomly they may be interpreted as symbols, in that they are effectively discrete because of the distribution of similarities of randomly chosen high-dimensional vectors. However, where the vectors to be composed are not chosen randomly the resultant similarity structure between the token vectors may carried over into the recursively composed structures, generalising them to continuous structures.

Smolensky's work covered both the mathematical structure of the representations and the operations that could be carried out on them. These operations included holistic transformations of recursive structures and multiple superposed structures. Thus, these structures may be transformed in a single step without sequentially unpacking them.

The beauty of tensor product binding compared to more typically connectionist techniques for producing structured representations (e.g. RAAM, Pollack [153]) is that iterative weight learning is not required. The binding properties arise directly from the architecture rather than an optimisation algorithm. New bindings may be constructed in a single time step and used immediately. This makes practical the creation and use of ephemeral cognitive structures.

I agree with Dave's second observation on the significance of Tony Plate's contribution. The major practical problems with Smolensky's tensor product binding are that the required vector dimension increases exponentially with the nesting depth of the structure to be represented and significant book-keeping is required to track the ordering of dimensions. Tony Plate overcame these problems with Holographic Reduced Representations (1994) [148]. HRRs can be viewed as compressed tensor products. They provide similar representational capabilities to tensor product binding, but have the remarkable practical advantage that a compositional structure occupies only as much representational space as an individual constituent. (Pollack's RAAM provides the same space saving, but at the cost of weight learning.)

To the best of my knowledge there are only a few people working actively with HRRs or closely related systems: Tony Plate, Pentti Kanerva, Ross Gayler, and Chris Eliasmith. Plate and Eliasmith work with HRRs. Kanerva's spatter-coding and Gayler's multiplicative binding are very closely related to each other and are members of the compressed tensor product family (as are HRRs). The apparently low rate of progress in this area is at least partly due to the low number of active researchers. However, I believe the area has gone (relatively) quiet because the research has moved into a qualitatively different phase. The work of Smolensky and Plate demonstrated that practical, connectionist, recursive structures are feasible. Now the focus has shifted to how these methods may be used to implement cognitive operations.

"No one is trying to build distributed connectionist reasoning systems any more, like the connectionist production system Touretzky and Hinton [200] built". This work was an important demonstration of technical capability and an implementation of a well understood symbolic architecture. The researchers working with the compressed tensor product family are attempting to develop connectionist architectures that take advantage of the unique strengths of the representations to implement cognitive operations. By definition, they are not seeking to implement known symbolic architectures. Thus, they have taken on the slow and difficult task of simultaneously developing novel models of cognitive processes and implementing them in novel ways.

Plate, Kanerva, and Gayler were all present at the Analogy'98 workshop in Sofia [58, 59, 98, 150]. They believe that analogy is central to human thought and that analogical mapping may be more easily implemented with compressed tensor product techniques than classical symbolic techniques because systematic substitution of terms is effectively a primitive operation in their systems. Eliasmith has worked on a distributed model of analogical mapping, reimplementing the Holyoak and Thagard [87] ACME model using HRRs . Accounts of Eliasmith's work are available from `http://ascc.artsci.wustl.edu/~ celiasmi`.

Returning to Touretzky's original question, this particular branch of connectionist symbol processing is not dead. It is in the early phase of striking out in a new direction.

Symbolic Methods in Neural Networks

C. Lee Giles

NEC Research Institute, USA, giles@research.nj.nec.com

What is often not realized is that symbolic methods, especially automata, and neural networks share a long history. The first work on neural networks and automata (actually sequential machines - automata implementations), was that of McCulloch and Pitts [131]. Surprisingly, it has also been referenced as the first paper on finite-state automata (FSA) [88], on Artificial Intelligence (Boden), and on recurrent neural networks (RNNs) [62]. In addition, the recurrent network (with instantaneous feedback) in the second part of this paper was then reinterpreted by Kleene [101] as a FSA in "Representation of Events in Nerve Nets and Finite Automata", published in the edited book "Automata Studies" by Shannon and McCarthy. Sometimes this paper is cited as the first article on finite state machines [143]. Minsky [134] discusses symbolic computation in the form of automata with neural networks both in his dissertation and in his book "Computation: Finite and Infinite Machines" which has a chapter on "Neural Networks: Automata Made up of Parts." All of the early work referenced above assumes that the neuron's activation function is hard-thresholded, not a "soft" sigmoid.

All of the early work on automata and neural networks was concerned with automata synthesis, i.e. how automata are built or designed into neural networks. Because most automata when implemented as a sequential machine requires feedback, the neural networks were necessarily recurrent ones (for an exception see Clouse [25]). It's important to note that the early work (with the exception of Minsky) did not often make a clear distinction between automata (directed, labeled, acyclic graphs [sometimes with stacks and tapes]) and sequential machines (logic and feedback delays) and was mostly concerned with FSA. There was little interest with the exception of Minsky in moving up the automata hierarchy to pushdown automata and Turing Machines.

One important area in which symbolic methods have recently been used in neural networks is that of representation, i.e. theoretically what computational structures neural networks are provably equivalent or not equivalent to. (The earliest recent work in the area seems to be Pollack [151].) Not surprisingly, all neural network architectures do not have the same computational power. Siegelmann [177, 176] proves that some recurrent network architectures are at least Turing equivalent while Frasconi [49] shows the computational limitations of a local RNN architecture. Giles [61] and Kremer [104] show that certain growing methods can be computationally limited. There has also been specific work on automata representation in neural networks, see for example, Casey [23] (convergence of FSA extraction), Omlin et al [138] (also a comparison of the complexity of many encoding methods and an extension of work by Alon [7]), Frasconi [50] (radial basis functions), Maass [122] (spiked neurons). This type of work is continuing in other computation structures, examples being graphs and tree grammars (Frasconi et al [51], Sperduti [187]) and is important for broadening the scope and power of neural computing systems.

"The Tragedy" of Connectionist Symbol Processing

Lev Goldfarb

University of New Brunswick, Canada, goldfarb@unb.ca

About 8-9 years ago, soon after the birth of the connectionists mailing list, there was a discussion somewhat related to the present one (about the state of connectionist symbol processing). I recall stating, in essence, that the "connectionist symbol processing" destroys most of the *class* information contained in the *symbolic* training set, *if the symbolic representation–and therefore the corresponding "symbolic operations"–are to be taken seriously.* This point appears to be not an obvious one. Why?

I believe that the situation is mainly explained by the fact that in mathematics the concept of symbolic *representation* has not yet been addressed, simply because no classical application areas have lead to it. Probably as a result, one "forgets" that the connectionist *representation space*–the vector space over the reals–*by its very definition* (via the two operations, vector addition and scalar multiplication) allows one "to see" only the compositions of the two mentioned operations and practically no *"symbolic operations"*, e.g. various deletion, insertion and substitution operations on strings.

From a formal point of view, it is important to keep in mind that each of the latter "symbolic operations" is a multivalued function defined on the set of strings over a finite alphabet (e.g. a may occur in many places in the same string).

In other words, if, for example, we encode the strings over a finite alphabet by vectors in a finite-dimensional vector space over the reals and try to recover any original symbolic "operation", e.g. "insertion of $abcacc$", using only the operations of the vector space or even, additionally, any fixed in advance finite set of non-linear functions, we see that without "cheating", i.e. without looking "back" at the symbolic operation itself, this is impossible.

The relevance of the last observation becomes more clear when, by analogy with the numeric case, one begins to view the inductive learning process in a symbolic environment as that of discovering the optimal (w.r.t. the training set) symbolic distance measure, where these distance measures are now defined via the *dynamically updated* set of weighted symbolic operations [64].

In such a "symbolic" framework, in contrast to the classical numeric framework, the *discovered* operations of weight 0, i.e. those corresponding to the some "important class features", induce on the symbolic space topology that is quite different from the unique topology of the finite-dimensional vector space.

In fact, it turns out that in a sense the symbolic representational bias is substantially more general than the numeric bias. Put differently, for practical purposes, the normed vector space is a very special case of the more general "symbolic space".

It should be pointed out that the basic "computer science" concept of abstract data type, or ADT, also strongly suggests to view the "differences" between the structured objects in term of the corresponding set of operations.

Some idea about the advantages of the "symbolic spaces" over the classical numeric spaces can, perhaps, be gleaned by comparing the known "numeric" solutions of the generalized parity problem, the problem quite notorious within the connectionist community, with the following, "symbolic", solution (the learning algorithm is omitted).

THE PARITY CLASS PROBLEM

The alphabet: $A = \{a, b\}$.

Input set S (i.e. the input space without the distance function):
The set of strings over A.

The parity class C: The set of strings with an even number of b's.

Example of a positive training set C^+:

$aababbbaabbaa$

$baabaaaababa$

$abbaaaaaaaaaaaaaa$

$bbabbbbaaaaabab$

aaa

Solution to the parity problem, i.e. inductive (parity) class representation:
One element from C^+, e.g. aaa, plus the following 3 weighted operations (note that the sum of the weights is 1)

deletion/insertion of a (weight 0)

deletion/insertion of b (weight 1)

deletion/insertion of bb (weight 0)

This means, in particular, that the distance function D between any two strings from the input set S is now defined as the shortest weighted path (generated by the above 3 operations) between these strings. The class is now defined as the set of all strings in the space (S, D) whose distance from aaa is 0.

On Connectionist Symbol Processing and the Question of A Natural Representational Ontology

Mitsuharu Hadeishi

Open Mind Research, mitsu@openmind.com

Some people claim that the fact that the input space to a connectionist or connectionist-style architecture can be represented as an n-tuple of values (which can be naturally interpreted as a Euclidean vector space) by itself provides sufficient evidence to shed significant doubt on the ability of connectionist architectures to develop or learn to deal with symbolic representations or symbolic problems. This is questionable for a variety of reasons, most notably because the input space may be mapped through what amounts to an arbitrary differentiable function approximator (i.e , a connectionist or connectionist-style architecture) [54, 89] which allow these kinds of systems to construct new models [154] and distort the original metric arbitrarily to bias it in favor of particular learning tasks [16, 17]. The learning process depends on the entire architecture of the network and the definition of the whole algorithm. In other words, the metric structure of the input space considered on its own does not need to be a significant factor when it comes to evaluating whether or not connectionism is capable of symbolic processing—in any case, one cannot evaluate the potential efficacy of the learning algorithm on the basis of just this. In addition, though fixed-dimensional, fixed-precision representations cannot handle arbitrarily recursive structures (with the point raised by Tony Plate that allowing bags of multiple fixed-dimensional, fixed-precision vector space representations such as his Holographic Reduced Representations [148, 145] *can* encode arbitrarily nested structure, and others have noted fixed-dimensional, infinite-precision vectors can be used in the same way, for example Jordan Pollack's RAAMs [152]), this does not prevent you from using sequential-in-time representations, for example, in conjunction with recurrent networks (which can emulate any Turing machine in theory [177]).

While using symbolic representations can certainly lead to faster solutions to symbolic problems [164], it is not clear that this is the most general way to approach all problems. In other words, the representation and the bias imposed by a given representation can lead to enhancements in the efficiency of a learning algorithm for a given problem domain, but a symbolic or any other specific representation is not necessarily an efficient representation for every problem domain—in fact, it seems obvious that this is not the case. In addition, one can conceive of connectionist and neural architectures which use both distributed and symbolic representations [212, 213]. Perhaps metaphorical reasoning requires this sort of flexibility (for example, attempting to understand poetic expressions).

In general, the notion that there is a single preferred ontology for all representational problems is a highly suspect one. While it is clear that symbolic ontologies (for example) have power (particularly due to their ability to be recursively recombined [148, 145, 152]), nevertheless they evolved over a very long time [128]. To say they are best a priori seems contradictory, since symbolic forms of representation have obviously not always existed and many other representations have also been used and are still being used, and there is no particular reason to think any specific representation is the "only" or "best" natural representation for all problems [82, 83]. While it is a seductive goal to try to find one ideal representation/ontology, the fact is that in nature, you can think of representations as stable basis elements used by feedback systems [14, 15] (i.e., during the interaction between an organism and its environment, stable patterns can evolve naturally—whatever works—as the organism interacts with its environment, or the organism attempts to regulate its own processes, these representations may be symbolic or not, as needed [137]). To speak of a single "best" representation is to ignore this fundamental aspect of what a representation is and to treat observations as though they were being carried out by some disembodied abstract entity that exists outside of a physical system or feedback loop: but there is no abstract "observation" or "observer", only the workings of a physical system [48]. Since there is no abstract entity called "an observation" being "made" by a disembodied "observer object", you have to look at the whole system (including both observer and observed) in order to interpret the representation (i.e., the information in the signal) whether the representation is symbolic, distributed, temporal, spatial, local, or something else. To be more concrete: a symbol on a page or an activation level of a neuron is meaningless in and of itself; it is only meaningful in the context of the system of which it is a part, and thus it is the properties of the whole system which determine the natural biases (and thus the bases) and generate the natural representation for the system.

Connectionist Research On The Problem Of Symbols: An Opportunity To Recast How Symbols Work By Reconsidering The Boundaries Of Cognitive Systems

Brian Hazlehurst

University of California, San Diego, USA, briahn@shn.net

In a recent response to David Touretsky's inquiry about the state of the art in connectionist symbol processing, Doug Blank raises some good points about cognition and how we should study it.

One I agree with is:

> Building and manipulating structured representations or binding variables via networks should not be our goals.

One I am neutral about is:

> Neither should creating a model such that we can understand its inner workings [be one of our goals].

Another I agree with is:

> Rather, we should focus on the techniques that allow a system to self-organize such that it can solve The Bigger Problems.

However, it is precisely awareness of The Bigger Problems that have focused people's attention on symbols and human abilities in processing symbols. I share the views expressed above, but also believe that the "problem of symbols" is core to The Bigger Problems. So what is a way around this apparent paradox?

One way is to recognize that "connectionist research on the problem of symbols" is not the same thing as "connectionist symbol processing". The latter type of research often casts the problem as one of mimicking the computational properties of symbols in a direct way: value-binding, compositional structure, systematicity, etc.

However, while each of these properties must hold at some level of description, and for some stage of development of the cognitive system, they need not apply at every level of description of that system nor for all time that the system exists. That is, these properties need not be atomic primitives of the system for them to play the roles that they do in the system.

For instance the properties of compositionality and systematicity so often used to describe natural language, are often taken as basic primitives of human language. Defining the system in this way leads us to build these properties into the primitives of our computational framework for modelling the phenomena – witness the Language of Thought model [47] which underlies the symbolic framework. (The framework, in turn, receives credit for being an "inspectable innner workings" methodology. A nice convenience perhaps, but certainly nature is not in the business of creating methodological conveniences.)

One real contribution of connectionism has been the introduction of a modelling framework which doesn't require us to make this commitment to building observed properties of the cognitive system into the primitives of the framework. By taking more seriously the micro-level structure and processing, we also have had an opportunity to reconstruct the appropriate boundaries of the cognitive system. In order to account for the properties of symbols and symbol-processing, I believe, this means considering the interactions among people and the environments they co-habit in performing cognitive work. In so reconstituting the boundary of the cognitive system, the required properties of symbols may well emerge from, rather than be primitives of, individual brains.

To return to the example of natural language, the challenge is to determine how agents who share a world and a set of tasks which must get accomplished in that world can employ a communication medium to facilitate the job. Furthermore, we wish to address the important question: can the medium support the complex properties found in natural language? Can such an arrangement, exhibiting the powers of symbol processing, exist without symbol systems being enscripted in the wetware?

Research addressing these questions is a line of work that Edwin Hutchins and I have pursued in collaboration for several years [93, 94, 74] . Our models entail demonstrations which answer these particular questions in the affirmative. For instance, we have shown how sharing of lexical items need not preceed the task of communication if the full cognitive system (vocal/gesture plus perception plus social commitment) is recognized and modelled. We have also demonstrated that a compositional communication system is readily achieved by agents which must negotiate the formation of the system, while individual verbalizers of inner organization fail to create compositional systems. This robust result follows from the fact that individual creators' verbalizations (in our simulations) never have to do any communicative work, and thus never get exploited in production of higher order structure – such as a grammar which serves to encourage expected constructions.

This work has affinities to other connectionist work in language acquisition [24] and symbol-grounding [72, 73, 36] , and is informed by a line of philosophical thinking about lanuguage which stems from Wittegenstein [105, 19, 45]. Our work has its roots in Cognitive Anthropology, where the organization of behavior is what needs to be explained and the basic data are the traffic in material exchanges (including language and gesture) which evoke and create meanings. However, given the broad array of documented ways in which organized behavior comes about through cultural processes, we pay attention (as do all anthropologists) to the inter-personal and historical aspects of cognitive systems.

To summarize, I agree with Doug Blank that connectionism needs to go beyond the framing of the agenda which has been inherited from symbolic theorizing and modelling. However, I also believe that crucial issues of The Big Problems will continue to be derived from the nature of symbols and symbol processing. My solution for how to meet both of these (seemingly at-odds) requirements lies with understanding that symbol processing is significantly informed by the domains of person-environment, person-person and across-generation phenomena.

Formal Semantic Model for Neural Networks

Michael J. Healy

The Boeing Company, University of Washington, USA, michael.j.healy@boeing.com

Understanding the semantic issues involved in symbolic processing with neural networks requires a mathematical approach, to disambiguate the terminology and understand the relationship between symbols, connectionist processing, and data. Rule extraction is one example of an active literature involving symbolic processing with neural networks. This work is mostly empirical, with formal modeling considered only infrequently (see [9] for an overview of the field and the issues most often addressed). A mathematical model for the semantics of symbolic processing with neural networks requires more than a stated symbolic representation of connectionist processing and connection weight memories: It requires an explicit semantic model as well. In such a model, the "universe" of things the symbolic concepts are about receives as much attention as the concepts themselves.

For example, if the concepts are descriptions of airplane part assemblies (fuselage, right engine, landing gear, landing gear strut, landing gear hydraulic fluid line, ...), then there are different examples of each concept—say, for a Boeing 747 or Airbus 320. An example can be represented in the mathematical model as a "point" in a space of airplane assemblies. Each point is associated with a set of concepts, representing what is known about that point.

A mathematical modeling effort in progress[75] has resulted in a statement about some important logical properties of rule bases in general and rule extraction with neural networks and other machine learning systems in particular. The main finding in this effort to date is that a sound and complete rule base—one in which the rules are actually valid for all the data and which has all the rules—has the semantics of a continuous function between topological spaces. The concepts of each space are associated with the open sets, and the things the concepts are about (such as the airplane assemblies in the example) are the points. The reference previously given discusses this, with some explanation. A more extensive development will be available in a forthcoming paper[76].

The mathematical semantic model referred to here is based upon geometric logic and its semantics. The initial version of the semantics is expressed in terms of point-set topology. This is the simple version; geometric logic is really a categorical logic, and category theory has a much greater expressive power. Geometric logic is very strict in what it takes to assert a statement. It is meant to represent observational statements, ones whose positive instances can be observed. Topology is the mathematical study of "nearness"; the connection with logic is that an instance of the "universe of discouse" satisfies a collection of logical statements that are related through logical inference; conversely, the instances of related statements satisfy a "similarity" or "nearness" relationship. Continuous functions between different universes (or sub-universes, called domains) preserve these similarity relationships. Continuity is really the mathematical way of saying "similar things map to similar things".

In terms of the previous example, in which airplane parts are points in a space, an important kind of similarity arises. To exemplify, one point might represent a left-side landing gear strut for a 320, but that might be all that is known about it. Alternatively, more could be known: It could be known that it comes from a particular airplane, "tail number such-and-such" (the airplane's unique ID is called its tail number). There is a partial order (a mathematical relation) in which a point representing an arbitrary airplane assembly is "less than" a point representing an arbitrary strut, which in turn is "less than" a point representing a 320 strut, and this in turn is "less than" a strut for any particular tail numbered 320. This order relation defines a specialization hierarchy: The higher you go in the order, the more specialized the points are—the more information they contain. Conversely, lower points in the hierarchy represent higher points that are similar in that they share some set of concepts.

The "universe" containing the points as elements is a topological space, and the lattice of open sets (ordered by set inclusion) of the topological space corresponds to the specialization order (which goes in the opposite direction). The logical entailment relation for the concepts corresponds directly to the subset relation. So, for example, the concept "landing gear strut" entails the concept "airplane assembly". Correspondingly, the set of all landing gear struts is a subset of the set of all airplane assemblies. Continuous functions preserve the ordering of points and the ordering of open sets as well.

This mathematical model relates directly to the work being done in rule extraction, even with the many different approaches and neural network models in use. The claim is that it supports the intuition that many researchers in the field seem to have about the instances of neural processing as rule instances. Another aspect of the model is that it begins to address semiotic issues—the relationship of sign-systems (e.g., symbol-systems) to semantics. Finally, the topological model is consistent with probabilistic modeling and, apparently, fuzzy logic.

Language Learning and Processing with Temporal Synchrony Variable Binding

Jamie Henderson

University of Exeter, UK, J.B.Henderson@exeter.ac.uk

Natural language, and particularly syntax, has traditionally been a bastion of symbolic processing. In recent years statistical methods have usurped some aspects of the symbolic approach, but one aspect which has not been successfully challenged is the use of structured representations. Thus, it still seems that the regularities in natural language can only be captured by dividing a sentence into discrete constituents and expressing generalizations in terms of these constituents. This poses a challenge to connectionism, but by using Temporal Synchrony Variable Binding (TSVB), we can represent both these constituents and these generalizations in a connectionist network.

TSVB is the method of using the synchrony of activation pulses to represent entities. This proposal was originally made on biological grounds ([205] and see [144]), and has also been justified on computational grounds [171]. A crucial characteristic of TSVB is that a single fixed set of link weights can be applied to multiple dynamically created entities. This characteristic allows a network's representation of syntactic regularities to capture generalizations across syntactic constituents [79]. While the class of generalizations that TSVB can capture is more constrained than the total set typically used in linguistic theories, they are sufficient to express the generalizations required for syntactic parsing, and these constraints even make significant linguistic predictions ([77], [78], [80]).

One advantage of connectionist representations over symbolic ones is that the ability to represent generalizations leads naturally to an ability to learn generalizations. For TSVB and syntactic parsing, this potential has been realized by extending the Simple Recurrent Network (SRN) architecture [43] with TSVB, producing an architecture we call Simple Synchrony Networks (SSNs) ([110], [81]). This architecture inherits from SRNs the ability to learn generalizations across positions in the input sequence. In addition, SSNs can not only represent syntactic constituency in their outputs, but they can learn generalizations across syntactic constituents. This generalization ability has been demonstrated through toy grammar experiments [109], and through the application of SSNs to learning broad coverage syntactic parsing from a corpus of naturally occurring text ([81], [110]). The success of these initial experiments bodes well for the future of this architecture, both within natural language processing and for a wide range of other complex domains.

A Connectionist Theory of Learning Proportional Analogies and the Nature of Associations Among Concepts

Nilendu G. Jani, Daniel S. Levine

University of Texas, Arlington, USA, nilendu_jani@yahoo.com,b344dsl@utarlg.uta.edu

A connectionist neural network has been developed that can simulate the learning of some simple proportional analogies. These analogies include, for example, a) red square: red circle :: yellow square: ????; b) apple: red :: banana:????; c) a:b:: s: ????.

Underlying the development of our analogy network is a theory for how the brain learns that pairs of concepts are associated in a particular manner. Traditional Hebbian learning of associations is necessary for this process but not sufficient. This is because Hebbian learning simply says, for example, that the concepts "apple" and "red" have been associated. It does not tell us the nature of the relationship that has been learned between "apple" and "red." The types of context-dependent interlevel connections in our network suggest a nonlocal type of learning that in some manner involves association among more than two nodes or neurons at once. Such connections have been called synaptic triads [40] and related to potential cell responses in the prefrontal cortex.

Some additional types of connections are suggested by the problem of modeling analogies. These types of connections have not yet been verified by brain imaging, but our work suggests that they may occur and, possibly, be made and broken quickly in the course of working memory encoding. For example, we include in our network various types of working memory connections that bear some kinship to what have been called differential Hebbian connections [102, 103]. In these connections, one can learn mappings between concepts such as "keep red the same"; "change red to yellow"; "turn off red"; "turn on yellow," et cetera. Also, we include a kind of weight transport (distantly analogous to what occurs in backpropagation networks) so that, for example, "red to red" can be transported to a different instance of color, such as "yellow to yellow." (In future modifications, we may also include transport "upward" and "downward" in the network to simulate property generalization and inheritance.)

Once a particular type of conceptual mapping has been learned between "apple" and "red" in our network, for example, this mapping can be applied to the concept of "banana" to generate "yellow." The network instantiation we are developing, based on common connectionist "building blocks" such as associative learning, competition, and adaptive resonance (see, e.g., [69, 115]), along with additional principles suggested by analogy data, is a step toward a theory of interactions among several brain areas to develop and learn meaningful relationships between concepts. This network is designed for a problem domain that is based on analogies among either relatively low-level, mundane percepts or abstract categories of such low-level percepts. Hence, at this stage it does not capture the type of high-level semantic analogies dealt with by the models of Hummel and Holyoak [91] and Plate [149] nor the analogical ambiguities obtained from detailed relationships among letters by the COPYCAT model of Mitchell [135]. We believe that the general conception of our model may subsequently be extendable to such high-level or ambiguous problem domains. However, we first wished to capture proportional analogies among low-level concepts and percepts by means of connectionist architectures that followed some biologically realistic principles of organization.

Grammar-based Neural Nets

Simon Lucas

University of Essex, UK, sml@essex.ac.uk

I would suggest that most recurrent neural net architectures are not fundamentally more 'neural' than hidden Markov models - we can think of an HMM as a neural net with second-order weights and linear activation functions.

HMMs are, of course, very much alive and kicking, and routinely successfully applied to problems in speech and OCR for example. It might be argued that the HMMs tend to employ less distributed representations than RNNs, but even if this is true, is it of practical significance?

There has been some interesting work exploring the links between HMMs and RNNs [22, 18].

Also related to the discussion is the Syntactic Neural Network (SNN) - an architecture I developed in my PhD thesis [120, 118].

The SNN is a modular architecture that is able to parse and (in some cases) infer context-free (and therefore also regular, linear etc.) grammars.

The architecture is composed of Local Inference Machines (LIMs) that rewrite pairs of symbols. These are then arranged in a matrix parser formation [219] to handle general context-free grammars - or we can alter the SNN macro-structure (i.e. the way in which individual LIMs are connected together) in order to specifically deal with simpler classes of grammar such as regular, strictly-hierarchical or linear. The LIM remains unchanged.

In my thesis I only developed a local learning rule for the strictly-hierarchical grammar, which was a specialisation of the inside/outside algorithm [11] for training stochastic context-free grammars.

By constructing the LIMs from forward-backward modules [119] however, any SNN that you construct automatically has an associated training algorithm. How well this works in practice has to be tested empirically. I've already proven this to work for regular grammars, and work is currently in progress to test other important cases.

What SNNs, Alpha-Nets and IOHMMs have in common is their explicit grammatical interpretation. Having trained the 'neural net', one can directly extract an equivalent grammar. This can be seen as advantageous from a practical engineering viewpoint, but also perhaps, makes these systems a bit less seductive from a connectionist viewpoint - one of the attractions of conventional RNNs has been their theoretical universal computing power - though in practice I'm not aware of any convincing demonstrations of them learning anything beyond regular languages (I would be happy to stand corrected on this point).

Representation, Reasoning and Learning with Distributed Representations

Tony Plate

Bios Group LP, USA, tplate@ibm.net

Work on higher-level connectionist processing has made steady progress over the last decade. Several solutions to some of difficult representational problems have been developed, and researchers are starting to build more complex systems.

There are three main problems concerning representation: binding, recursive structure, and learning. Interesting techniques have been developed for solving all of these problems individually. However, no one has yet built a connectionist system which can solve all of these problems at once and learn underlying recursive structure from a raw input data stream.

Binding and Structure

There are two broad families of solutions to the problems of binding and representing recursive structure: static conjunctive codes [84, 179, 145] , and binding by temporal synchrony [171, 91]. Both are promising and presently have different strengths: conjunctive codes such as Holographic Reduced Representations (HRRs) cope with recursive structure in a more uniform fashion, while systems based on temporal synchrony have exhibited more advanced processing (e.g., LISA). I'll restrict my comments mainly to conjunctive codes.

Regarding the problems of binding and structure in static distributed representations, Mitsu Hadeishi expressed a claim that used to be widely held:

> "An input space which consists of a fixed number of dimensions cannot handle recursive combinations"

A number of people have shown that it is possible to represent arbitrarily nested concepts in space with a fixed number of dimensions. Furthermore, the resulting representations have interesting and useful properties not shared by their symbolic counterparts. Very briefly, the way one can do this is by using vector-space operations for addition and multiplication to implement the conceptual operations of forming collections and binding concepts. For example, one can build a distributed representation for a shape configuration#33 of "circle above triangle" as:

$$\text{config33} = \text{vertical} + \text{circle} + \text{triangle} + \text{ontop}*\text{circle} + \text{below}*\text{triangle}$$

where each symbol here is a vector. By using an appropriate multiplication operation (in my HRRs [145] I used circular, or wrapped, convolution), the reduced representation of the compositional concept (e.g., config33) has the same dimension as its components, and can readily be used as a component in other higher-level relations. Decoding is accomplished easily using inverse operations. Large recursive structures can be chunked into smaller parts so that arbitrary precision in vectors is not required. To do this requires some additional machinery: an auto-associative clean-up memory in which all chunks and sub-chunks can be stored. Quite a few people have devised representational schemes based on some form of conjunctive coding (though few have used chunking), e.g., Smolensky's Tensor Products [179], Pollack's (1990) RAAMs [153], Sperduti's (1994) LRAAMs [186], Kanerva's (1996) Binary Spatter Codes [97], and Gayler's (1998) Braid operator [58]. Another related scheme that uses distributed representations and tensor product bindings (but not role-filler bindings) is Halford, Wilson and Philips' STAR model [71].

Some of the useful properties of HRRs and these types of conjunctive representations in general are as follows:

1. The reduced, distributed representation (e.g., config33) functions like a pointer, but is more than a mere pointer in that information about it contents is available directly without having to "follow" the "pointer." This makes it possible to do some types processing without having to unpack the structures.

2. The vector-space similarity of representations (i.e., the dot-product) reflects both superficial and structural similarity of structures.

3. There are fast, approximate, vector-space techniques for doing "structural" computations like finding corresponding objects in two analogies, or doing structural transformations.

4. HRRs and binary spatter codes are fully distributed in the sense usefully defined by Bryan Thompson: "the equi-presence of the encoding of an entity or compositional relation among all elements of the representation." In HRRs everything is represented over all of the units. Suppose one has a vector X which represents a certain structure. Then just the first half of X will also represent that whole structure, though it will be noisier.

5. HRRs scale well to representing large numbers of entities and relations. The vector dimensionality required is high – in the hundreds to thousands of elements. But, HRRs have an interesting scaling property – toy problems involving a just a couple dozen relations might require a dimensionality of 1000, but the dimensionality doesn't need to increase much (to 2 or 4 thousand) to handle problems involving tens of thousands of relations.

The way these distributed representations work almost always involves superimposing vectors which represent different concepts, e.g., different role filler bindings (in HRRs) or different propositions (in STAR, Halford et al [71]). In this respect, the following point made by Jerry Feldman may seem puzzling:

> "Parallel Distributed Processing is a contradiction in terms. To the extent that representing a concept involves all of the units in a system, only one concept can be active at a time."

However, while this claim is true as it stands, it is often not applicable in connectionist systems. This is because distributed representations are usually have a high level of redundancy: all of the units may be involved in representing a concept, but none are essential. Thus, one can easily represent more than one concept at a time in most distributed representation schemes. Part of the beauty of distributed representations is the soft limit on the number of concepts that can be represented at once. This limit depends on the dimensionality of the system, the redundancy in representations, the similarity structure of the concepts, and so forth. And of course one can also have different modules within a system. But, the important point here is that even within a single PDP module, one can still represent (and process) multiple concepts at once.

Learning

Even if one has a way of constructing distributed representations of complex structures by systematic composition of lower level concepts, one still needs to be able to develop good distributed representations for the lower level concepts. These representations should reflect the similarity: similar concepts should be represented by similar vectors. One method for learning such representations is Latent Semantic Analysis (LSA) [39] , also known as Latent Semantic Indexing (LSI). LSA is a method for taking a large corpus of text and constructing vector representations for words in such a way that similar words are represented by similar vectors. LSA works by representing a word by its context (harkenning back to a comment made by Firth (1957): "You shall know a word by the company it keeps"), and then reducing the dimensionality of the context using singular value decomposition (SVD). (For those familiar with principal component analysis, the reduction is essentially the same as PCA.) The vectors constructed by LSA can be of any size, but it seems that moderately high dimensions work best: 100 to 300 elements. It turns out that one can do all sorts of surprising things with these vectors. One can construct vectors which represent documents and queries by merely summing the vectors for their words and then do information retrieval by finding the document with a vector most similar to the vector for the query. This automatically gets around the problem of synonyms, since synonyms tend to have similar vectors. One can do the same thing with multiple-choice tests, by forming vectors for the question and each potential answer, and choosing the answer with the most similar vector to the question. Landauer et al [108] describe a system built along these lines can pass first-year psychology exams and TOEFL tests. It is intriguing that all of these results are achieved by treating texts as unordered bags of words – there are no complex reasoning operations involved. LSA could provide an excellent source of representations for use in a more complex connectionist systems (using connectionist in a very broad sense here). One attractive property of LSA is that it is fast enough that it can be used on many tens of thousands of documents to derive vectors for many thousands of words. This is exiting because it could allow one to start building connectionist systems which deal with full-range vocabularies and large varied task sets (as in info. retrieval and related tasks), and which do more interesting processing than just forming the bag-of-words content of a document a la vanilla-LSA.

Processing: Complex Reasoning

As mentioned by Ross Gayler and others, analogy processing is a very promising area for the application of connectionist ideas. There are a few reasons for analogy being interesting: it appears to be widespread in human cognition, structural relationships are important to the task, no explicit variables need be involved, and rule-based reasoning can be seen as a very specialized version of the task. One very interesting model of analogical processing with (partially) distributed representations is Hummel and Holyoak's LISA model [91]. This model uses distributed representations for roles and fillers, binding them together with temporal synchrony, and achieves quite impressive results. Similar models based on static conjunctive binding codes have not yet been demonstrated, though powerful primitive operations for manipulating structures and finding corresponding objects have been identified [146].

The Input Space Is The Real World

George Reeke

The Rockefeller University, USA, reeke@lobimo.rockefeller.edu

Mitsu Hadeishi wrote:

> "The point I am making is simply that after one has transformed the input space, two points which begin 'close together' (not infinitesimally close, but just close) may end up far apart and vice versa. The mapping can be degenerate, singular, etc. Why is the metric on the initial space, then, so important, after all these transformations? Distance measured in the input space may have very little correlation with distance in the output space."

I can't help stepping in with the following observation. The reason that distance in the input space is so important is that the input space is the real world. It is generally (not always, of course) useful for biological organisms to make similar responses to similar situations–this is what we call "generalization". For this reason, whatever kind of representation is used, it probably should not distort the real-world metric too much. It is perhaps too easy when thinking in terms of mathematical abstractions to forget what the purpose of all these transformations might be.

Mitsu Hadeishi replied:

> "Quite true, which is why I mentioned 'not infinitesimally close' since clearly the transformations need to be stable (for the most part) over small variations in the input (i.e., not have discontinuous variations). However, it can be useful for the transformations to have 'relatively sharp' boundaries, i.e., so as to be able to discriminate categories (consider a network whose output is deciding whether a letter presented to it is 'A' or 'B'—usually you would want the output to be varying relatively sharply between the categories.)"

The idea that relatively sharp boundaries should be introduced by input space transformations as a means of discriminating categories, and thus of defining symbols, is, in my opinion, at the heart of a fundamental problem with the symbol-processing approach to cognition. (This is an issue for any symbol-processing approach, not just for connectionist symbol processing, of course.) We do indeed need symbols to perform certain higher functions that involve logic, but I do not think we use them for ordinary perceptual categorization or even to define natural language categories. In abstract symbol systems, the boundaries between symbols are set in advance or at any rate remain fixed during any one set of transactions. In human cognitive systems, the boundaries can change according to context and can even be different in two consecutive sentences that use the same terms. There is only minimal confusion because the world is right there to act as a reference to disambiguate the categories via linkages detectable in perceptual mappings. This is what makes it possible for a small set of words to cover the enormously varied set of microscopically different situations that arise in the real world. It is what enables generalization to work across even rather distant categories without inducing confusion in other circumstances where the boundaries need to be different. This is not a property of the kind of mathematical transformations that Mitsu and others in this discussion have been talking about. I claim that the kind of fixed symbols needed for logical reasoning can be built "on top" of flexible perceptual categories tied to the ever-changing real world, but it is not possible to build the flexible perceptual categories "on top" of fixed abstract symbols, because the glue of the distance metric in the real world is lost in an abstract symbol system.

Gerald Edelman some time ago introduced the term "zoomability" to refer to the property by which neural representations of the world contain at any time numerous links at multiple scales to other chunks of the world [41]. He and I have discussed in several places [156, 42, 157] why we believe that perceptual mappings must possess this and related properties in order to support human categorization, language, and reasoning.

Learning and Reasoning with Connectionist Representations

Dan Roth

University of Illinois, Urbana-Champaign, USA, danr@cs.uiuc.edu

Learning and reasoning have been long recognized as fundamental phenomena of intelligence. Much progress has been made in understanding them from a computational perspective. Unfortunately, the theories that have emerged for the two are somewhat disparate. If one wants to develop computational models that account for the flexibility, adaptability and speed of reasoning, a central consideration is how the knowledge is acquired and how this process, of interaction with its environment, influences the performance of the reasoning system. Developing a unifying theory for the Learning and Reasoning phenomena, we believe, is currently the central theoretical question we face if we want to make significant progress towards understanding how the brain or a man-made machine perform knowledge intensive inferences.

An attempt to cast research in this direction as the study of symbolic processes using connectionist representations is, we believe, misguided. It focuses the research on presenting traditionally studied knowledge representations and inference problems using a connectionist architecture, neglecting that (1) this by itself does not resolve the intractability of reasoning with these representations and (2) the fact that connectionist representations are not inherently more robust than other equivalent representations; non-brittleness is a virtue of the way representations are acquired. On the learning side, it emphasizes an unnatural separation of learning algorithms (all function approximation algorithms) to "symbolic" and "non-symbolic".

Our work in this direction has concentrated on developing the theoretical basis within which to address some of the obstacles and on developing an experimental paradigm within which realistic experiments (in terms of scale and resources) can be performed to validate the theoretical basis. In particular, we have developed a connectionist architecture and algorithmic tools that have already shown superior results on some large-scale, real-world inference problems in the natural language domain.

The formal framework developed to address such questions is the *learning to reason* (L2R) approach – an integrated theory of learning, knowledge representation and reasoning. Within L2R it is shown [100] that through interaction with the world, the system truly gains additional reasoning power over what is possible in the traditional setting, where the inference problem is studied independently of the knowledge acquisition stage. In particular, cases are presented where *learning to reason* is feasible but either reasoning from a given representation or learning these representations do not have efficient solutions. A suggestion of how to use L2R within a connectionist architecture is developed in [161]. The specific implementation used there utilizes a model-based approach to reasoning [99] and yields a network that can support instantaneous deduction and abduction, in cases that are intractable using other knowledge representations. This is achieved by interpreting the connectionist architecture as encoding *examples* acquired via interaction with the environment, and allows for the integration of the inference and learning processes.

This framework is being studied within Valiant's Neuroidal paradigm [203], a computational model that is intended to be consistent with the gross biological constraints we currently understand. This is a programmable model which makes minimal assumptions about the computing elements. It is composed of two types of units: *circuit units* and *image units* with the intention that a network of circuit units will form the long term memory of the system while image units can be thought of as the working memory of the system [204]. It is equipped with on-line learning mechanisms and a decision support mechanism.

Our experimental system, *SNOW*, is influenced by the Neuroidal model, and has already been shown to perform remarkably well on several real-world natural language inferences tasks such as context sensitive word correction (Spell), part-of-speech tagging (POS) and prepositional phrase attachment [65, 163, 106]. The *SNOW* (Sparse Network of Winnows) architecture is a network of threshold gates utilizing the Winnow [117] learning algorithm as an update rule. The system consists of a very large number of items which correspond to high-level concepts, for which humans have words, as well as lower-level predicates from which the high-level ones are composed. Lower-level predicates encode aspects of the current state of the world, and are input to the architecture from the outside. The high-level concepts are learned as functions of the lower-level predicates. When learning from text, for example, complex features of pre-defined form (e.g., conjunctions of primitive features that appear in close proximity) that are observed in the text are allocated nodes in the network, and learning is done in terms of these complex features. This allows one to learn higher-than-linear representations using a learning algorithm that learns linear functions. This process becomes feasible computationally due to modest dependence of the algorithm used on the dimensionality of the domain. Learning in *SNOW* [162] proceeds in an on-line fashion, where every example (input sentence) is treated autonomously by (possibly) many target subnetwork. For example, in Spell, target nodes represent members of the confusion sets (e.g. {*desert*, *dessert*}); in POS, target nodes correspond to different pos tags. An example may be treated as a positive one for a few of the nodes and negative to others. At decision time given an input sentence which activates a subset of the input nodes, the information propagates through all the subnetworks and a learned decision support mechanism takes affect. Winnow, a local, mistake driven on-line learning algorithm, is used at each target node to learn its dependence on other nodes. Its key feature is that the number of examples it requires to learn the target node grows linearly with the number of *relevant* attributes and only logarithmically with the total number of attributes – allowing for a large feature set.

The system uses large text corpora to efficiently learn a network architecture consisting of hundreds of thousands of features. The learned network is able to create a useful level of knowledge and successfully perform several inference tasks. In this way, the system already addresses a few of the important computational issues that arise in learning to perform large-scale language inferences. Studying the interaction between subnetworks and devising binding mechanisms as part of a concrete implementation of the image units are some of the issues that are currently under investigation.

Understanding how the brain can perform knowledge intensive inferences such as language understanding, high level vision and planning and behave robustly when presented with previously unseen situations, is one of the great intellectual problems of our time. We have argued that the key to make progress in this direction is to develop unified theories of learning and reasoning, and reported on some theoretical progress in this direction. As important is the development of an experimental paradigm, so that progress can be measured using large scale experiments, with a methodology that is common in other sciences. We have presented the *SNOW* system as an example for a successful system, capable of performing large real-world inferences.

SHRUTI – A Neurally Motivated Model of Rapid Relational Processing

Lokendra Shastri [1]

International Computer Science Institute, Berkeley, California, shastri@icsi.berkeley.edu

In order to understand language, a hearer must draw inferences to establish referential and causal coherence, generate expectations, and recognize speaker's intent. Yet we can understand language at the rate of several hundred words per minute. This suggests that we are capable of performing a wide range of inferences rapidly, spontaneously and without conscious effort – as though such inferences are a reflex response of our cognitive apparatus. In view of this, such reasoning may be described as *reflexive* reasoning [171].

This remarkable human ability poses a central challenge for cognitive science and computational neuroscience: How can a system of simple and slow neuron-like elements represent a large body of systematic knowledge and perform a wide range of inferences with such speed?

In 1989, V. Ajjanagadde and L. Shastri proposed a structured connectionist[2] model SHRUTI [1], which attempted to address this challenge and demonstrated how a network of slow neuron-like elements could encode a large body of structured knowledge including specific as well an instantiation independent knowledge, and perform a variety of inferences within a few hundred milliseconds [5][170][6][165][171]. D.R. Mani made several important contributions to the model [126] and also implemented the first SHRUTI simulator.

[1] Research on SHRUTI has been supported by: NSF grants SBR-9720398 and IRI 88-05465, ONR grant N00014-93-1-1149, DoD grant MDA904-96-C-1156, ARO grants DAA29-84-9-0027 and DAAL03-89-C-0031, DFG grant Schr 275/7-1, ICSI general funds, and subcontracts from Cognitive Technologies Inc. under ONR grant N00014-95-C-0182 and ARI contract DASW01-97-C-0038. Work on the CM-5 version of Shruti was supported by NSF Infrastructure Grant CDA-8722788.

[2] For an overview of the structured connectionist approach and its merits see [46][166]

SHRUTI suggested that the encoding of relational information (frames, predicates, etc.) is mediated by neural circuits composed of *focal clusters* and the dynamic representation and communication of relational *instances* involves the transient propagation of *rhythmic* activity across these clusters. A role-entity binding is represented within this rhythmic activity by the *synchronous* firing of appropriate cells. Systematic mappings — and other rule-like knowledge — are encoded by high-efficacy links that enable the propagation of rhythmic activity across focal clusters, and a fact in long-term memory is a temporal pattern matcher circuit that fires when the static bindings it encodes match the dynamic bindings encoded in the rhythmic activity propagating through the neural circuitry.

SHRUTI showed that reflexive reasoning can be the spontaneous and natural outcome of a neurally plausible system. In SHRUTI there is no separate interpreter or inference mechanism that manipulates and rewrites symbols. The network encoding of commonsense knowledge is a vivid *model* of the agent's environment and when the nodes in this model are activated to reflect a given state of affairs in the environment, the model spontaneously *simulates* the behavior of the external world and in doing so finds coherent explanations and makes predictions [171].

SHRUTI also identified a number of constraints on the representation and processing of relational knowledge and predicted the capacity of the dynamic memory underlying reflexive reasoning. On the basis of neurophysiological data pertaining to occurrence of synchronous activity in the γ band, SHRUTI lead to the prediction that a large number of facts can be active simultaneously and a large number of rules can fire in parallel during an episode of reflexive reasoning. However, the number of distinct entities participating as role-fillers in this activity must remain very small (≈ 7).

The possible role of synchronous activity in dynamic neural representations had been suggested by other researchers (e.g., Malsburg [206]), but SHRUTI provided a detailed account of how synchronous activation can be harnessed to solve problems in the representation and processing of high-level (symbolic) conceptual knowledge. Several other models that use synchrony to solve the dynamic binding problem have since been proposed (e.g., [92][91]) and there has emerged a rich body of neurophysiological evidence suggesting that synchronous activity might indeed play an important role in neural information processing (e.g., [178][202]).

The relevance of SHRUTI extends beyond reasoning to other forms of rapid processing of relational information. For example, J. Henderson [77] has shown that constraints predicted by SHRUTI explain several properties of language processing including garden path effects and our limited ability to deal with center-embedding.

Over the past several years, SHRUTI has been augmented in a number of ways [168] in collaborative work between Shastri, his students, and others. These enhancements enable SHRUTI to:

1. Encode negated facts and rules and deal with inconsistent beliefs (see [172])
2. Seek coherent explanations for observations
3. Encode soft/evidential rules and facts (with D.J. Grannes and C. Wendelken)
4. Represent types and instances, and the subtype and supertype relations in an effective manner and support limited forms of quantification [169]
5. Instantiate entities dynamically during reasoning (with D.J. Grannes and J. Hobbs)
6. Represent multiple-consequent rules (with D.J. Grannes and C. Wendelken)
7. Exhibit priming effects (with D.J. Grannes, B. Thompson, and C. Wendelken)
8. Support context-sensitive unification of entities (with B. Thompson and C. Wendelken)
9. Tune network weights and rule-strengths via supervised learning (with D.J. Grannes, B. Thompson, and C. Wendelken)
10. Realize control and coordination mechanisms required for encoding parameterized schemas and reactive plans (see [173]).

The above enhancements have been incorporated into the SHRUTI simulator by D.J. Grannes and C. Wendelken. In ongoing work, with M.S. Cohen, B. Thompson, and C. Wendelken, the author is augmenting SHRUTI to integrate the propagation of *belief* with the propagation of *utility*. This integration would allow SHRUTI's sense of utility to direct its search for answers and explanations. In addition to the developments mentioned above, V. Ajjanagadde has worked on the problem of abductive reasoning and pursued an alternate set of representational mechanisms [3][4].

Applications

SHRUTI (circa 1993) has been mapped onto the CM-5 by D.R. Mani [125]. The resulting system can encode knowledge bases with over 500,000 (randomly generated) rules and facts, and yet respond to a range of queries requiring derivations of depth five in under 250 milliseconds. Even queries with derivation depths of eight are answered in well under a second. The mapping of SHRUTI onto a network/cluster of workstations is currently under investigation.

The SHRUTI model meshes with the NTL project at ICSI [2] on language acquisition and provides connectionist solutions for several representational and computational issues arising in the project. We are also engaged in the integration of SHRUTI and its *reflexive* capabilities with a metacognitive (reflective) component in collaboration with B. Thompson and M.S. Cohen. The reflective component is responsible for directing the focus of attention, making and testing assumptions, identifying and responding to conflicting interpretations and/or goals, locating unreliable conclusions, and managing risk (see [32] and B. Thompson and M.S. Cohen in this issue).

INFERNET:A Model of Deductive Reasoning

Jacques Sougne

University of Liège, Belgium, J.Sougne@ulg.ac.be

INFERNET is a model of deductive reasoning. It uses a distributed network of spiking nodes. Variable binding is achieved by temporal synchrony. While it is not a new technique (see Grossberg & Somers [70]; Hummel & Holyoak [91]; Lisman & Idiart [116]; Luck & Vogel [121], Usher & Donnelly [202]; Shastri & Ajjanagadde [171]), the use of distributed representation, and the way it solves the problem of multiple instantiation is new.

INFERNET is able to deal with conditional reasoning (material implication) including negated conditional [182, 180]. Here are the 16 forms:

$A \supset B$, A;	$A \supset B$, $\neg A$;	$A \supset B$, B;	$A \supset B$, $\neg B$
$A \supset \neg B$, A;	$A \supset \neg B$, $\neg A$;	$A \supset \neg B$, $\neg B$;	$A \supset \neg B$, B
$\neg A \supset B$, $\neg A$;	$\neg A \supset B$, A;	$\neg A \supset B$, B;	$\neg A \supset B$, $\neg B$
$\neg A \supset \neg B$, $\neg A$;	$\neg A \supset \neg B$, A;	$\neg A \supset \neg B$, $\neg B$;	$\neg A \supset \neg B$, B

The INFERNET simulator performance fit human data which are sensitive to negation and especially to double negations. This effect of negation is often referred as negative conclusion bias in the psychological literature.

INFERNET has also been applied on problem requiring multiple instantiations (Sougne [183, 184, 181]; Sougne & French [185]). In INFERNET, multiple instantiation is achieved by using the neurobiological phenomena of period doubling. Nodes pertaining to a doubly instantiated concept will sustain two oscillation. This means that these nodes will be able to synchronize with two different set of nodes. This method puts constraints on the treatment of multiple instantiation. The INFERNET simulator performance seems to fit human data for problems requiring multiple instantiation like:

```
Mark loves Helen and Helen loves John. Who is jealous of whom?
```

Due to distributed representation, INFERNET is sensitive to similarity of the concepts used in deductive tasks which are confirmed by empirical evidences [184, 185]). There is also an interesting effect of noise [184]. When white noise is added in the system (and if it is not too important) the performance of the system is improved. This phenomenon is known as Stochastic resonance (Levin & Miller [114]).

Localist Connectionist Models For Symbolic Processing

Ron Sun

The University of Alabama, USA, rsun@cs.ua.edu

There has been a variety of work in developing *localist* connectionist models for symbolic processing, as pointed out by messages posted on the connectionist mailing list by, e.g., Jerry Feldman and Lokendra Shastri. Although it has been discussed somewhat, a more detailed list of work in this area is still needed. The work in this field spans a large spectrum of application areas in AI and cognitive science. Thus, it seems to be reasonable and useful to summarize the work in terms of these areas.

- *Reasoning*, which includes work on commonsense reasoning, logic reasoning, case-based reasoning, reasoning based on schemas (frames), and so on. In terms of commonsense reasoning, we have the following work: Lange and Dyer [111], Shastri and Ajjanagadde [171], Sun [191], and Barnden [12]. These existing models take different approaches: Shastri and Ajjanagadde [171] and Sun [190] took a logic approach, while Barnden and Srinivas [13] took a case-based reasoning approach. Lange and Dyer [111] used reasoning based on given frames that contain world knowledge encoded in localist connectionist networks. Lacher et al [107] implemented MYCIN type reasoning with confidence factors for expert system applications.

- *Natural Language Processing*, including both syntactic and semantic processing. There are many models, including the following pieces of work: Henderson [77], Bookman [21], Regier [158], Wermter et al [216] and Bailey et al [10].

- *Learning of Symbolic Knowledge*, based on first learning in neural networks and then extracting symbolic knowledge. Such work can be justified as follows: In many instances, extraction of knowledge from neural networks is better than learning symbolic knowledge directly using symbolic algorithms, in algorithmic/computational or cognitive modeling terms. There are voluminous publications on this. For example, the most important pieces of work are Fu [53], Giles and Omlin [63], and Towell and Shavlik [201]. Some of these models involve somewhat distributed representation, but that's not the point. The point is that localist models can be trained and symbolic knowledge can be extracted from it. See also Sun et al [194] for a model in which neural networks and extracted symbolic rules work together to produce synergistic results.

- *Recognition and Recall*, including a variety of models such as Jacobs and Grainer [95, 96], Page and Norris [139], and the now classic McClelland and Rumelhart [129].

- *Memory*, which is an area where many models are being developed and applied by the psychology and cognitive science communities. See Hintzman [85] for a review.

- *Skill Learning*, including Sun and Peterson [195] and Sun et al [194], as well as some on-going projects that I am personally aware of (but, unfortunately, have no publications yet).

A question that naturally arises is: why should we use connectionist models (especially localist ones) for symbol processing, instead of symbolic models? There are so many different reasons. I cannot even begin to enumerate all the rationales for using localist models for symbolic processing discussed in these afore-cited pieces of work. These reasons may include:

1. Localist connectionist models are an apt description framework for a variety of cognitive processing. (See Grainger & Jacobs [68] for further information.)

2. The inherent processing characteristics of connectionist models (such as similarity-based processing, which can also be explored in localist models) make them suitable for cognitive processing.

3. Learning processes can naturally be applied to localist models (as opposed to learning LISP code), such as gradient descent, EM, etc. (As has been pointed out by many recently, localist models share many features with Bayesian networks. This actually has been recognized very early on, see for example, Sun [188, 189], in which a localist network is defined from a collection of hidden Markov models, and the Baum-Welch algorithm was used in learning.)

Finally, here are two more more bibliographical notes.

1. Sun and Bookman [193] contains a detailed annotated bibliography on high-level connectionist models that covers all the important work up to 1993.

2. In addition, see also the recently published edited collection, Sun and Alexandre [192], for further information.

In sum, the field of connectionist symbolic processing is alive and well, especially localist approaches, which have been reviewed here. Progresses are being made steadily, albeit slowly. There are reasons to expect further significant progress to be made, given all the afore-mentioned reasons for such models.

Stefan Wermter

Hybrid Neural Symbolic Agent Architectures based on Neuroscience Constraints

University of Sunderland, Sunderland, UK, stefan.wermter@sunderland.ac.uk

Adaptive symbolic and neural agents have received a lot of interest for different tasks, for instance speech/language integration and image/text integration [159, 155, 132, 216, 44, 220]. Hybrid neural symbolic methods have been shown to be able to reach a level where they can actually be further developed in real-world scenarios. A combination of symbolic and neural agents is possible in various *hybrid processing architectures*, which contain both symbolic and neural agents appropriate according to a specific task, e.g. integrating speech, text and images.

From the perspective of knowledge engineering, hybrid symbolic/neural agents are advantageous since different mutually complementary properties can be combined. Symbolic agents have advantages with respect to easy interpretation, explicit control, fast initial coding, dynamic variable binding and knowledge abstraction. On the other hand, neural agents show advantages for gradual analog plausibility, learning, robust fault-tolerant processing, and generalization to similar input. Since these advantages are mutually complementary, a hybrid symbolic neural architecture can be useful from the perspective of knowledge engineering if different processing strategies have to be supported.

A *loosely coupled hybrid architecture* has separate symbolic and neural agents. The control flow is sequential in the sense that processing has to be finished in one agent before the next agent can begin. Only one agent is active at any time, and the communication between agents is unidirectional. An example architecture where the division of symbolic and neural work is loosely coupled has been described in a model for structural parsing within the SCAN framework [212]. A *tightly coupled hybrid architecture* contains separate symbolic and neural agents, and control and communication are via common internal data structures in each agent. The main difference between loosely and tightly coupled hybrid architectures is common data structures which allow a bidirectional exchange of knowledge between two or more agents.

In an *integrated hybrid architecture* there is no discernible external difference between symbolic and neural agents, since the agents have the same interface and they are embedded in the same architecture.

The control flow may be parallel and the communication between symbolic and neural agents is via messages. Communication may be bidirectional between many agents, although not all possible communication channels have to be used. One example of an integrated hybrid architecture was developed for exploring integrated hybrid processing for spontaneous spoken language analysis [214, 217, 215].

SCREEN is an integrated architecture since it does not crucially rely on a single neural or symbolic agent. Rather, there are many neural agents, but also some symbolic agents. They have a common interface and they can communicate with each other in many directions. From an agent-external point of view it does not matter whether the internal processing within an agent is neural or symbolic. This architecture therefore exploits a full integration of symbolic and neural processing at the agent level. Furthermore, the agents can run in parallel and produce an analysis in an incremental manner. While integrated hybrid architectures provide the current state of the art for neural symbolic interaction [217], there is still a static way of communication between a static number of agents. We are working towards a new different class of hybrid dynamic architectures which can be used for different multimodal scenarios.

However, for building more sophisticated intelligent computational agents in the long run, principles from cognitive neuroscience also have to be considered for building neural architectures which are based more on the brain functions. For instance using whole-brain fMRI, it seems possible that activation patterns of distributed regions in the brain serve as associative memory during visual-verbal, episodic declarative memory tasks. Such encoding and retrieval experiments could be evidence for the impact of medial parietal regions in storage and recall of declarative memory. Principles of neuroscience and brain function could have an important influence on building complexer well-grounded neural network systems, in particular for areas like associative memory storage and retrieval, intelligent navigation, speech/language processing, or vision. However, besides the integration of neuroscience constraints into artificial neural networks it will also be important to understand the symbolic interpretation of neural networks at higher cognitive levels since the human real neural networks are capable of dealing with symbolic processes.

Recursive Computation in a Bounded Metric Space

Whitney Tabor

University of Connecticut, USA, tabor@cs.cornell.edu

Tony Plate's inspiring work on Holographic Reduced Representations (HRRs) [148, 145] breaks new ground by providing a mathematically well-grounded method of encoding tree-structured objects in fixed-width vectors. Following related work by Smolensky on tensor-product representations [179], Plate's analysis of the saturation properties of the codes helps provide new insight into the representational differences between standard symbolic computers and connectionist networks. But the saturation analysis speaks in terms of typical properties of HRR vectors and does not give explicit insight into the geometric properties of particular encodings.

A related line of work [196, 136] focuses on designing the geometry of specific trajectories of metric space computers, including many connectionist networks. One essential idea (proposed in embryonic form in Pollack [151, 152], Siegelmann and Sonntag [175], Wiles and Elman [218], Rodriguez, Wiles, and Elman, to appear [160]) is to use fractals to organize recursive computations in a bounded metric space. Cris Moore [136] provides the first substantial development of this idea, relating it to the traditional practice of classifying machines based on their computational power. He shows, for example, that every context free language can be recognized by some "dynamical recognizer" that moves around on an elaborated, one-dimensional Cantor Set. I have described a similar method which operates on high-dimensional Cantor sets and thus leads to an especially natural implementation in neural hardware [196, 197].

This approach sheds some new light on the relationship between conventional and connectionist computation by showing how we can use the structured entities recognized by traditional computational theory (e.g., particular context free grammars) as bearing points in navigating the larger set [136, 174] of computing devices embodied in many analog computers. A useful future project would be to to use this kind of computational/geometric perspective to interpret HRRs and related outer product representations.

Naturalistic Decision Making and Models of Computational Intelligence [3]

Bryan B. Thompson, Marvin S. Cohen

Cognitive Technologies, Inc., USA, {bryan,mcohen}@cog-tech.com

We are working with an cognitive model of the acquisition and use of decision-making skills. It is a naturalistic approach, which begins with the way experienced, effective decision makers actually make decisions in real-world tasks [28],[27],[26]. The model is based on interviews with 14 tactical Naval officers [34] and 33 U.S. Army command staff [29]. We have found (as have others in expert-novice research) that these officers use methods that combine pattern recognition with strategies for effectively facilitating recognition, verifying its results, and constructing more adequate models when recognition fails. From this foundation, we have developed an adaptive model of decision making that integrates recognition and metacognitive processes. Such metacognitive processes, once acquired, may also support more rapid (reflexive) explanation-based learning of domain knowledge. We call this the Recognition / Metacognition (R/M) model [27],[34],[33].

This research combines (i) a theory of human decision making, the R/M model; (ii) empirical research testing that model and its training implications in several domains, including tactical Navy, Army battlefield and commercial aviation decision making [31],[33],[30],[52]; (iii) research on connectionist architectures for model-based reinforcement learning [198],[209],[210],[211]; and (iv) development of a connectionist model for rapid recognition-based domain reasoning (`Shruti`, [171],[167], which is described by Shastri elsewhere in this survey).

We are currently engaged in computational modeling of reflexive (recognitional) and reflective (metacognitive) behaviors. To this end, we have used a structured connectionist model of inferential long-term memory, with good results. The reflexive system is based on the `Shruti` model proposed by Shastri and Ajjanagadde [171],[167]. Working with Shastri, we have extended the model to incorporate supervised learning, priming, and other features. We are currently working on an integration of belief and utility within this model. The resulting network will be able to not only reflexively construct interpretations of evidence from its environment, but will be able to reflexively plan and execute responses at multiple levels of abstraction as well.

This reflexive system, implemented with `Shruti`, is coupled to a metacognitive system, which is responsible for directing the focus of attention, making and testing assumptions, identifying and responding to conflicting interpretations of evidence and/or goals, locating unreliable conclusions, and managing risk. In essence, the metacognitive system is a meta-controller that learns behaviors which manipulate the context under which recognitional processing occurs. While the recognitional system uses a causal model of the world, the metacognitive system represents the evidence-conclusion relationships, or arguments [199], that have been instantiated in the world model during recognitional processing. That is, it has a second order causal representation of patterns of activation in the world model. According to Pennington and Hastie [142],[141] and others (e.g., Pearl, [140]), decision makers use causal representations when data are voluminous, complex, and interdependent. It is no surprise that causal models are suggested at both the recognitional and metacognitive level of analysis.

The significance of this research lies in what it can tell us about the structure and dynamics of human long-term memory (LTM). There is strong cognitive evidence for distinct recognitional and metacognitive processes. In addition, there is independent evidence concerning the computational complexity of reflexive inference, as well as limits on biologically plausible computation, that constrain viable models of human cognition. Such limits on recognitional resources necessitate adaptive attention shifting behaviors, which extend the scope of reflexive inference. Complex *metacognitive* skills are developed from these simple attention shifting mechanisms. Such skills provide for reasoning explicitly about uncertainty. They are used to identify

[3] This work has been supported by the Office of Navy Research (N00014-95-C-0182) and by the Army Research Institute (DASW01-97-C-0038)

incomplete arguments, unreliable assumptions, and conflicting goals or evidence, and then reframe the recognitional problem by changing assumptions and/or the focus of attention. When stakes and time warrant more than reflexive behavior, real decision-makers use such skills to evaluate and improve interpretations and plans.

By combining a reflexive reasoner with attention shifting mechanisms, we identify systems capable of assembling larger, coherent interpretations and plans, and of reasoning explicitly about uncertainty, rather than simply aggregating alternative explanations. Research and experiments with the computational model, and a wealth of empirical decision-making studies [29],[34], suggest that the development of an executive attention function, such as the proposed metacognitive system, may be necessary for, and integral to, the development of structured, inferential LTM. Further, these data suggest that *skilled* LTM, that is, expertise, is developed through the application of such an executive attention function.

For instance, in an incident described in an interview with a tactical Navy officer, a Libyan gunboat turned toward a U.S. Navy cruiser in Libyan-claimed waters, and increased its speed. This stereotypical attack pattern suggested to the officer that the crew of the gunboat had hostile intentions. The Captain, however, shifted his attention from the kinematic cues that had convinced the junior officer, and as a result identified a series of problems in the junior officer's assessment. First, the Captain identified gaps in the argument for hostile intent, i.e., there was no account of how the gunboat detected own ship, why own ship was selected as a target, and why the gunboat was selected as an attack platform. As it turned out, the gunboat was not thought to have the capability to localize the cruiser at the distance at which it had turned toward own ship. Moreover, the gunboat already had passed closely to another U.S. ship that would have been a more accessible and equally lucrative target. Finally, the gunboat itself was not an effective platform against a cruiser. These considerations conflicted with the original assessment of hostile intent. The Captain proceeded to consider and evaluate assumptions that might explain the discrepant evidence (e.g., that the gunboat had found own ship as a target of opportunity). He also attempted to construct and evaluate a plausible story on the assumption that the gunboat did not have hostile intent. Finally, he carefully estimated the time available before the risk from the gunboat would become unacceptable, and spent that time examining possible explanations for the presence and behavior of the gunboat, alternatively identifying and testing assumptions. One of the distinguishing traits of expertise is *how* such knowledge becomes connected *such that* relevant evidence is automatically integrated and uncertainties identified. (E.g., Both (a) "The gunboat lacked the means to localize the cruiser", and (b) "It was an ineffective threat" provide arguments against *hostile-intent*).

Taken together, such research and data reveal new computational forms for cognitive systems. In current research, we are developing a new method for Approximate Dynamic Programming, termed *causal-HDP*. Dynamic programming is one of the few methods available for optimizing behavior over time in stochastic environments. Approximate solutions to Dynamic Programming, such as Heuristic Dynamic Programming (HDP, Werbos [208],[209]) and Q-Learning (Watkins, [207]), have received a great deal of attention both in neuro-control systems, known as Adaptive Critics, and in genetic algorithms. The basic expression of dynamic programming is the Bellman equation [90], by which the expected future utility of present state, and an optimal policy function (behavior), may be computed, e.g.:

$$\mathsf{J}^*(\underline{\mathsf{R}}(\mathsf{t})) = \mathsf{Max}_{\underline{\mathsf{u}}(\mathsf{t})}\langle \mathsf{U}(\mathsf{t}) + \mathsf{J}^*(\underline{\mathsf{R}}(\mathsf{t}+1))\rangle \tag{1}$$

where J^* is the cumulative expected future value, $\underline{u}(t)$ is a vector of actions at time t, where the angle brackets denote the expected value, U is a utility function, and $\underline{R}(t)$ is the internal state of the reasoning agent at time t. Methods such as Q-Learning, while widely used, are unable to exploit a world model. This rather severe limitation is overcome by model-based adaptive control methods, such as HDP.

Existing methods estimate the Bellman equation by incrementally "backing-up" estimates of future rewards (from $t+1$ to time t) to revise an evaluation function. The evaluation function takes shape as the agent repeatedly re-visits states, and is used to revise a policy function that implements the behavior of the agent. The policy function probabilistically chooses actions that maximize expected value given the current estimate provided by the evaluation function. As a result, the evaluation function updates slowly and behavioral changes lag even further behind. Existing methods can easily "crystallize" on an evaluation function, thereby locking in a behavior. If the model or environment subsequently change, the evaluation and policy functions often must undergo catastrophic re-learning or remain markedly sub-optimal.

Rather than directly "looking-up" the evaluation of a state within an associative memory, we are exploring methods that reflexively compute an n-step approximation of the evaluation function from the evidence available, the current assumptions, including the focus of attention, the domain knowledge encoded in the world model, and estimates of expected future value and prior-probability that lie beyond the scope of reflexive inference; that is, at the dynamically determined edges of the belief network. Estimates of expected future value and prior-probability are associatively stored at all variables (whether simple truth values, predicates, or consequents and antecedents of rules) in the belief network. Whenever the propagation of activation does not reach beyond a variable (due to the dynamic limits on reflexive processing), the prior-probability (for causes) or the expected future value (for effects) substitutes for further computation. We utilize the computational structure of causal models within `Shruti` to directly compute these multi-step approximations of the Bellman equation during reflexive reasoning. This approach is a hybrid between classical time-backward dynamic programming and incremental, single-step, heuristic dynamic programming methods. It responds immediately, and optimally, to changes in the world model *as long as* the relevant information lies within the scope of reflexive inference, and it performs no worse when relying on estimates not within the current focus of attention.

This computational structure provides a principled method by which an agent may reflexively evaluate situations, plan, and take actions. The result is a new method for model-based adaptive control suitable for high-level cognitive problems. Agents using such mechanisms (attention shifting, approximate dynamic programming, and a structured, inferential LTM) will be able to operate at multiple levels of spatial and temporal abstraction, resulting in non-linear increases in planning horizons. If borne out by future research, this work will have a profound impact on how we model intelligent behavior and learning, and on what we can hope to accomplish with such models.

The perspective that only a fully distributed representation qualifies as a connectionist solution to structure is, perhaps, what warrants challenge. In this research with Shastri, we demonstrate that inferential structure and variable binding may be addressed within a connectionist model that respects, and is motivated by, cognitive, computational, and biological constraints. The brain is by no measure without internal structure on both gross and very detailed levels. While research has yet to identify rich mechanisms for dealing with structured representations and inference within fully distributed representations, it also has yet to fully explore the potential of specialized and adaptive neural structures for systematic reasoning. We suggest that there may be rich rewards in both pursuits.

References

[1] –. http://www.icsi.berkeley.edu/~ shastri/shruti/index.html.

[2] –. http://www.icsi.berkeley.edu/NTL.

[3] V. Ajjanagadde. Abductive reasoning in connectionist networks: Incorporating variables, background knowledge, and structured explanada. Technical report, WSI 91-6, Wilhelm-Schickard Institute, University of Tubingen, Tubingen, Germany, 1991.

[4] V. Ajjanagadde. *Rule-Based Reasoning in Connectionist Networks.* PhD thesis, University of Minnesota, 1997.

[5] V. Ajjanagadde and L. Shastri. Efficient inference with multi-place predicates and variables in a connectionist. In *Proceedings of the Eleventh Conference of the Cognitive Science Society*, pages 396–403, 1989.

[6] V. Ajjanagadde and L. Shastri. Rules and variables in neural nets. *Neural Computation*, 3:121–134, 1991.

[7] N. Alon, A.K. Dewdney, and T.J. Ott. Efficient simulation of finite automata by neural nets. *Journal of the Association for Computing Machinery*, 38(2):495–514, 1991.

[8] R. Andrews, R. Cable, J. Diederich, S. Geva, M. Golea, R. Hayward, C. Ho-Stuart, and A. Tickle. An evaluation and comparison of techniques for extracting and refining rules from artificial neural networks. Technical report, QUT NRC Technical Report, 1996. http://sky.fit.qut.edu.au/NRC/ftpsite/QUTNRC-96-01-04.html.

[9] R. Andrews, J. Diederich, and A.B. Tickle. Survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, 8(6):373–389, 1995.

[10] D. Bailey, J. Feldman, S. Narayanan, and G. Lakoff. Embodied lexical development. In *Proceedings of the Nineteenth Annual Meeting of the Cognitive Science Society COGSCI-97*. Stanford, Stanford University Press, 1997.

[11] J.K. Baker. Trainable grammars for speech recognition. In D.H. Klatt and J.J. Wolf, editors, *Speech Communication Papers for the 97th Meeting of the Acoustical Society of America*, pages 547 – 550. Algorithmics Press, New York, (1979).

[12] J. Barnden. Complex symbol-processing in conposit. In R. Sun and L. Bookman, editors, *Architectures incorporating neural and symbolic processes*, volume 1. Kluwer, 1994.

[13] J. Barnden and K. Srinivas. Overcoming rule-based rigidity and connectionist limitations through massively parallel case-based reasoning. *International Journal of Man-Machine Studies*, 1992.

[14] G. Bateson. *Steps to an Ecology of Mind*. Ballantine, New York, 1972.

[15] G. Bateson. *Mind and Nature - A Necessary Unity*. Bantam Books, New York, 1980.

[16] J. Baxter. *Learning Internal Representations*. PhD thesis, The Flinders University of South Australia, 1995.

[17] J. Baxter. The canonical distortion measure for vector quantization and function approximation. In Sebastian Thrun and Lorien Pratt, editors, *Learning to Learn*, pages 159–179. Kluwer Academic Publishers, 1998.

[18] Y. Bengio and P. Frasconi. Input-output HMMs for sequence processing. *IEEE Transactions on Neural Networks*, 7:1231 – 1249, (1996).

[19] M. Bickhard. The social nature of the functional nature of language. In M. Hickman, editor, *Social and functional approaches to language and thought*, pages –. Academic Press, Orlando, FL, 1987.

[20] D.S. Blank. *Learning to see analogies: a connectionist exploration*. PhD thesis, Indiana University, Bloomington, 1997. http://dangermouse.uark.edu/thesis.html.

[21] L. Bookman. A framework for integrating relational and associational knowledge for comprehension. In R. Sun and L. Bookman, editors, *Architectures incorporating neural and symbolic processes*, volume 1. Kluwer, 1994.

[22] J.S. Bridle. Alpha-nets: a recurrent 'neural' network architecture with a hidden Markov model interpretation. *Speech Communication*, 9:83 – 92, (1990).

[23] M.P. Casey. The dynamics of discrete-time computation, with application to recurrent neural networks and finite state machine extraction. *Neural Computation*, 8(6):1135–1178, 1996.

[24] Y. Chauvin. *Symbol acquisition in humans and neural (PDP) networks*. PhD thesis, University of California, San Diego, 1988.

[25] D.S. Clouse, C.L. Giles, B.G. Horne, and G.W. Cottrell. Time-delay neural networks: Representation and induction of finite state machines. *IEEE Transactions on Neural Networks*, 8(5):1065, 1997.

[26] M.S. Cohen. The bottom line: naturalistic decision aiding. In G.A. Klein, J. Orasanu, R. Calderwood, and C.E. Zsambok, editors, *Decision making in action: Models and methods*, pages 265–269. Ablex Publishing Corporation, Norwood, NJ, 1993.

[27] M.S. Cohen. The naturalistic basis of decision biases. In G.A. Klein, J. Orasanu, R. Calderwood, and C.E. Zsambok, editors, *Decision making in action: Models and methods*, pages 51–99. Ablex Publishing Corporation, Norwood, NJ, 1993.

[28] M.S. Cohen. Three paradigms for viewing decision biases. In G.A. Klein, J. Orasanu, R. Calderwood, and C.E. Zsambok, editors, *Decision making in action: Models and methods*, volume 1, pages 36–50. Ablex Publishing Corporation, Norwood, NJ, 1993.

[29] M.S. Cohen, L. Adelman, M.A. Tolcott, T.A. Bresnick, and F. Marvin. A cognitive framework for battlefield commanders' situation assessment, October 1993. Arlington, VA: Cognitive Technologies, Inc.

[30] M.S. Cohen and J.T. Freeman. Thinking naturally about uncertainty. In *Proceedings of the Human Factors & Ergonomics Society, 40th Annual Meeting*, volume 1, pages 179–183, Santa Monica, CA, 1996. Human Factors and Ergonomics Society.

[31] M.S. Cohen, J.T. Freeman, and B.B. Thompson. Training the naturalistic decision maker. In C. Zsambok & G. Klein, editor, *Naturalistic decision making*, pages 257–268. Lawrence Erlbaum, Mahwah, NJ, 1997.

[32] M.S. Cohen, J.T. Freeman, and B.B. Thompson. Critical thinking skills in tactical decision making: A model and a training method. In J. Canon-Bowers and E. Salas, editors, *Decision-Making Under Stress: Implications for Training & Simulation*, volume 1. American Psychological Association Publications, Washington, DC, 1998. in press.

[33] M.S. Cohen, J.T. Freeman, and B.B. Thompson. Critical thinking skills in tactical decision making: A model and a training strategy. In Eduardo Cannon-Bowers, Janis A. & Salas, editor, *Decision making Under Stress: Implications for Training and Simulation*, pages 155–189. American Psychological Association, Washington, D.C., 1998.

[34] M.S. Cohen, J.T. Freeman, and S. Wolf. Meta-recognition in time stressed decision making: Recognizing , critiquing, and correcting. *Human Factors*, 38(2):206–219, 1996.

[35] M. Coltheart, B. Curtis, P. Atkins, and M. Haller. Models of reading aloud: Dual-route and parallel-distributed processing approaches. *Psychological Review*, 100:589–608, 1993.

[36] G. Cotrell, B. Bartell, and C. Haupt. Grounding meaning in perception. In *Proceedings of the 14th German Workshop on Artificial Intelligence. (GWAI-90)*, pages 308–321, Berlin, 1990. Springer-Verlag.

[37] M.W. Craven and J.W. Shavlik. Using sampling and queries to extract rules from trained neural networks. In *Machine Learning: Proceedings of the Eleventh International Conference*, 1994.

[38] F. de Saussure. Course in general linguistics, 1974.

[39] S. Deerwester, S. T. Dumais, T. K. Landauer, G. W. Furnas, and R. A. Harshman. Indexing by Latent Semantic Analysis. *Journal of the Society for Information Science*, 41(6):391–407, 1990.

[40] S. Dehaene, J.-P. Changeux, and J.P. Nadal. Neural networks that learn temporal sequences by selection. *Proceedings of the National Academy of Sciences*, 84:2727–2731, 1985.

[41] G.M. Edelman. *The Remembered Present.* Basic Books, New York, 1989.

[42] G.M. Edelman and G.N. Reeke. Is it possible to construct a perception machine? *Proceedings of the American Philosophical Society*, 134(1):36–73, 1990.

[43] J. L. Elman. Finding structure in time. *Cognitive Science*, 14(2):179–212, 1990.

[44] J.L. Elman, E.A. Bates, M.H. Johnson, A. Karmiloff-Smith, D. Parisi, and Plunkett. *Rethinking Innateness.* MIT Press, Cambridge, MA, 1996.

[45] C. Ernelling. *Understanding language acquisition.* SUNY Press, Albany, NY, 1993.

[46] J.A. Feldman. Neural representation of conceptual knowledge. In P. Culicover & R.M. Harnish L. Nadel, L.A. Cooper, editor, *Neural Connections, Mental Computation*, volume 1, pages 68–103. MIT Press, Cambridge, MA, 1989.

[47] J. Fodor. *The Language of Thought.* Harvard Press, Cambridge, MA, 1975.

[48] J.V. Francisco. Autopoiesis and a biology of intentionality. In *Proceedings of the Workshop Autopoiesis and Perception*, Dublin, 1992. Dublin City University, August.

[49] P. Frasconi and M. Gori. Computational capabilities of local-feedback recurrent networks acting as finite-state machines. *IEEE Transactions on Neural Networks*, 7(6), 1996. 1521-1524.

[50] P. Frasconi, M. Gori, M. Maggini, and G. Soda. Representation of finite state automata in recurrent radial basis function networks. *Machine Learning*, 23:5–32, 1996.

[51] P. Frasconi, M. Gori, and A. Sperduti. A framework for adaptive data structures processing. *IEEE Transactions on Neural Networks*, 9(5), 1998. 768.

[52] J. Freeman, M.S. Cohen, and B.T. Thompson. Time-stressed decision-making in the cockpit. In *Proceedings of the Association for Information Systems*, Baltimore, MD, 1998. Americas Conference. Published on CD-ROM.

[53] L.M. Fu. Rule learning by searching on adapted nets. In *Proc.of AAAI'91*, pages 590–595, 1991.

[54] K. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2:183–192, 1989.

[55] B.M. Garner. *A sensitivity analysis for symbolically trained multilayer adaptive feedforward neural networks.* PhD thesis, RMIT, 1997.

[56] B.M. Garner. A symbolic solution for adaptive feedforward neural networks found with a new training algorithm. In *Proc. ICCIMA'98*, pages 410–415, 1998.

[57] B.M. Garner. A training algorithm for adaptive feedforward neural networks that determines its own topology. In *Proc. ACNN'98*, pages 292–296, 1998.

[58] R.W. Gayler. Multiplicative binding, representation operators, and analogy. In *In K. Holyoak, D. Gentner, & B. Kokinov (Eds.) Advances in analogy research: Integration of theory and data from the cognitive, computational, and neural sciences*, page 405, Sofia, Bulgaria, 1998. New Bulgarian University. Poster abstract. Full poster available from the computer science section of http://cogprints.soton.ac.uk/abs/comp/199807020.

[59] R.W. Gayler and R. Wales. Connections, binding, unification, and analogical promiscuity. In *In K. Holyoak, D. Gentner, & B. Kokinov (Eds.) Advances in analogy research: Integration of theory and data from the cognitive, computational, and neural sciences*, pages 181–190, Sofia, Bulgaria, 1998. New Bulgarian University. Paper available from http://cogprints.soton.ac.uk/abs/comp/199807018. Accompanying poster available from http://cogprints.soton.ac.uk/abs/comp/199807019.

[60] D. Gentner and K. Forbus. Mac/fac: a model of similarity-based access and mapping. In *Proceedings of the Thirteenth Annual Cognitive Science Conference*, pages 504–509. Lawrence Erlbaum, 1991.

[61] C.L. Giles, D. Chen, G.Z. Sun, H.H. Chen, Y.C. Lee, and M.W. Goudreau. Constructive learning of recurrent neural networks: Limitations of recurrent casade correlation and a simple solution. *IEEE Transactions on Neural Networks*, 6(4):829–836, 1995.

[62] C.L. Giles, G.M. Kuhn, and R.J. Williams. Dynamic recurrent neural networks: Theory and applications. *IEEE Transactions on Neural Networks*, 5(2):153–156, 1994.

[63] C.L. Giles and C.W. Omlin. Extraction, insertion and refinement of production rules in recurrent neural networks. *Connection Science*, 5(3&4):307–, 1993. Special Issue on 'Architectures for Integrating Symbolic and Neural Processes'.

[64] L. Goldfarb and J. Hook. Why classical models for pattern recognition are not pattern recognition models. In Sameer Singh, editor, *Proc. International Conference on Advances in Pattern Recognition*, pages 405–414, London, 1998. Springer.

[65] A. R. Golding and D. Roth. A Winnow-based approach to spelling correction. *Machine Learning*, 1999. to appear in Special issue on Machine Learning and Natural Language.

[66] M. Golea. On the complexity of rule-extraction from neural networks and network-querying. In R. Andrews and J. Diederich, editors, *Rules and Networks*, pages 51–59. QUT Publication, Brisbane, 1996.

[67] J. Grainger and A.M. Jacobs. Orthographic processing in visual word recognition: A multiple read-out model. *Psychological Review*, 103:518–565, 1996.

[68] J. Grainger and A.M. Jacobs. *Localist connectionist approaches to human cognition*. Erlbaum, Mahwah, NJ, 1997.

[69] S. Grossberg. *Neural Networks and Natural Intelligence*. MIT Press, Cambridge, MA, 1988.

[70] S. Grossberg and D. Somers. Synchronized oscillations for binding spatially distributed feature codes into coherent spatial patterns. In G. Carpenter and S. Grossberg, editors, *Neural Networks for Vision and Image Processing*, volume 1, chapter 14, pages 385–405. MIT Press, Cambridge, 1992.

[71] G. Halford, W.H. Wilson, and S. Phillips. Processing capacity defined by relational complexity: Implications for comparative, developmental, and cognitive psychology. Behavioral and Brain Sciences, to appear.

[72] S. Harnad. The symbol grounding problem. *Physica D*, 42:335–346, 1990.

[73] S. Harnad. Connecting object to symbol in modelling cognition. In A. Clark and R. Lutz, editors, *Connectionism in context*, pages –. Springer-Verlag, New York, 1992.

[74] B. Hazlehurst and E. Hutchins. The emergence of propositions from the co-ordination of talk and action in a shared world. *Language and Cognitive Processes*, 13(2/3):373–424, 1998.

[75] M.J. Healy. Continuous functions and neural network semantics. *Nonlinear Analysis*, 30(3):1335–1341, 1997. Proc. of Second World Cong. of Nonlinear Analysts (WCNA96).

[76] M.J. Healy. A topological semantics for rule extraction with neural networks. *Connection Science*, page to appear, 1999.

[77] J. Henderson. Connectionist syntactic parsing using temporal variable binding. *Journal of Psycholinguistic Research*, 23(5):353–379, 1994.

[78] J. Henderson. *Description Based Parsing in a Connectionist Network*. PhD thesis, University of Pennsylvania, Philadelphia, PA, 1994. Technical Report MS-CIS-94-46.

[79] J. Henderson. A connectionist architecture with inherent systematicity. In *Proceedings of the Eighteenth Conference of the Cognitive Science Society*, pages 574–579, La Jolla, CA, 1996.

[80] J. Henderson. Constituency, context, and connectionism in syntactic parsing. In Matthew Crocker, Martin Pickering, and Charles Clifton, editors, *Architectures and Mechanisms for Language Processing*. Cambridge University Press, Cambridge UK, in press.

[81] J. Henderson and P. Lane. A connectionist architecture for learning to parse. In *Proceedings of COLING-ACL*, pages 531–537, Montreal, Quebec, Canada, 1998.

[82] F. Heylighen. Selection criteria for the evolution of knowledge. In *Proc. 13th Int. Congress on Cybernetics*, page 529, Namur, 1993. Association Internat. de Cybernetique.

[83] F. Heylighen. Objective, subjective and intersubjective selectors of knowledge. *Evolution and Cognition*, 3(1):63–67, 1997.

[84] G.E. Hinton. Implementing semantic networks in parallel hardware. In G. E. Hinton and J. A. Anderson, editors, *Parallel Models of Associative Memory*. Hillsdale, NJ: Erlbaum, updated edition, 1989.

[85] L. Hintzman. A review of models of memory. *Annual Review of Psychology*, 1996.

[86] D. Hofstadter and the Fluid Analogies Research Group. *Fluid concepts and creative analogies*. Basic Books, New York, NY, 1995.

[87] K.J. Holyoak and P. Thagard. Analogical mapping by constraint satisfaction. *Cognitive Science*, 13:295–355, 1989.

[88] J.E. Hopcroft and J.D. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley Publishing Company, Inc., Reading, MA, 1979.

[89] K.M. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[90] R. Howard. *Dynamic Programming and Markov Processes*. MIT Press, Cambridge, 1960.

[91] J. E. Hummel and K. J. Holyoak. Distributed representations of structure: A theory of analogical access and mapping. *Psychological Review*, 104(3):427–466, 1997.

[92] J.E. Hummel and I. Biederman. Dynamic binding in a neural network for shape recognition. *Psychological Review*, 99:480–517, 1982.

[93] E. Hutchins and B. Hazlehurst. Learning in the cultural process. In C. Langton, C. Taylor, D. Farmer, and S. Rasmussen, editors, *Artificial Life II.*, pages 689–706. Addison-Wesley, Redwood City, CA, 1991.

[94] E. Hutchins and B. Hazlehurst. How to invent a lexicon: The development of shared symbols in interaction. In N. Gilbert & R. Conte, editor, *Artificial Societies: The computer simulation of social life.* UCL Press, London, 1995.

[95] A.M. Jacobs and J. Grainger. Testing a semistochastic variant of the interactive activation model in different word recognition experiments. *Journal of Experimental Psychology: Human Perception and Performance*, 18:1174–1188, 1992.

[96] A.M. Jacobs and J. Grainger. Models of visual word recognition: Sampling the state of the art. *Journal of Experimental Psychology: Human Perception and Performance*, 20:1311–1334, 1994.

[97] P. Kanerva. Binary spatter-coding of ordered k-tuples. In C. von der Malsburg, W. von Seelen, J.C. Vorbruggen, and B. Sendhoff, editors, *Artificial Neural Networks–ICANN Proceedings*, volume 1112 of *Lecture Notes in Computer Science*, pages 869–873, Berlin, 1996. Springer.

[98] P. Kanerva. Dual role of analogy in the design of a cognitive computer. In *In K. Holyoak, D. Gentner, & B. Kokinov (Eds.) Advances in analogy research: Integration of theory and data from the cognitive, computational, and neural sciences*, pages 164–170, Sofia, Bulgaria, 1998. New Bulgarian University. Copies of the proceedings may be ordered from: analogy@cogs.nbu.acad.bg.

[99] R. Khardon and D. Roth. Reasoning with models. *Artificial Intelligence*, 87(1-2):187–213, 1996.

[100] R. Khardon and D. Roth. Learning to reason. *Journal of the ACM*, 44(5):697–725, 1997.

[101] S.C. Kleene. Representation of events in nerve nets and finite automata. In C.E. Shannon and J. McCarthy, editors, *Automata Studies*, pages 3–42. Princeton University Press, Princeton, N.J., 1956.

[102] A.H. Klopf. A neuronal model of classical conditioning. *Psychobiology*, 16:85–125, 1988.

[103] B. Kosko. Differential Hebbian learning. In J. S. Denker, editor, *Neural Networks for Computing: AIP Conference Proceedings.*, pages 265–270. American Institute of Physics, New York, 1986.

[104] S.C. Kremer. Finite state automata that recurrent cascade-correlation cannot represent. In D. Touretzky, M. Mozer, and M. Hasselno, editors, *Advances in Neural Information Processing Systems 8*. MIT Press, 1996. 612-618.

[105] S. Kripke. *Wittgenstein on rules and private language.* Harvard Press, Cambridge, MA, 1982.

[106] Y. Krymolowski and D. Roth. Incorporating knowledge in natural language learning: A case study. In *COLING-ACL'98 workshop on the Usage of WordNet in Natural Language Processing Systems*, pages 121–127, 1998.

[107] R. Lacher, S. Hruska, and D. Kunciky. Backpropagation learning in expert networks. Technical report, 91-015, Florida State University, Florida, 1992. also in IEEE TNN.

[108] T. K. Landauer, D. Laham, and P. W. Foltz. Learning human-like knowledge with singular value decomposition: A progress report. In *Neural Information Processing Systems (NIPS*97)*, 1998.

[109] P. Lane. *Simple Synchrony Networks: A New Connectionist Architecture Applied to Natural Language Parsing.* PhD thesis, University of Exeter, Exeter, UK, in preparation.

[110] P. Lane and J. Henderson. Simple synchrony networks: Learning to parse natural language with temporal synchrony variable binding. In *Proceedings of the International Conference on Artificial Neural Networks*, pages 615–620, Skövde, Sweden, 1998.

[111] T. Lange and M. Dyer. High level inferencing in a connectionist network. *Connection Science*, pages 181–217, 1989. See also Lange's chapter in Sun and Bookman (1994).

[112] E. Leach. *Culture and Communication.* Cambridge University Press, 1976.

[113] C. Levi-Strauss. *Structural Anthropology.* Allen Lane, 1968.

[114] J.E. Levin and J.P. Miller. Broadband neural encoding in the cricket cercal sensory system enhanced by stochastic resonance. *Nature*, 380:165–168, 1996.

[115] D.S. Levine. *Introduction to Neural and Cognitive Modeling.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1991.

[116] J.E. Lisman and M.A.P. Idiart. Storage of 7 2 short-term memories in oscillatory subcycles. *Science*, 267:1512–1515, 1995.

[117] N. Littlestone. Learning quickly when irrelevant attributes abound: A new linear-threshold algorithm. *Machine Learning*, 2:285–318, 1988.

[118] S.M. Lucas. Connectionist architectures for syntactic pattern recognition. *PhD Thesis, University of Southampton*, (1991).

[119] S.M. Lucas. Forward-backward building blocks for evolving neural networks with intrinsic learning behaviours. In *Lecture Notes in Computer Science (1240): Biological and artificial computation: from neuroscience to technology*, pages 723 – 732. Springer-Verlag, Berlin, (1997).

[120] S.M. Lucas and R.I. Damper. Syntactic neural networks. *Connection Science*, 2:199 – 225, (1990).

[121] S.J. Luck and E.K. Vogel. Response from luck and vogel. *Trends in Cognitive Sciences*, 2(3):78–80, 1998.

[122] W. Maass and P. Orponen. On the effect of analog noise in discrete-time analog computations. *Neural Computation*, 10:1071–1095, 1998.

[123] F. Maire. A partial order for the m-of-n rule-extraction algorithm. *IEEE Transactions on Neural Networks*, 8(6):1542–1544, 1997.

[124] F. Maire. Rule-extraction by backpropagation of polyhedra, 1998. Submitted to Neural Networks.

[125] D.R. Mani. *The Design and Implementation of Massively Parallel Knowledge Representation and Reasoning Systems: A Connectionist Approach.* PhD thesis, University of Pennsylvania, 1995.

[126] D.R. Mani and L. Shastri. Reflexive reasoning with multiple-instantiation in in a connectionist reasoning system with a typed hierarchy. *Connection Science*, 5(3&4):205–242, 1993.

[127] D. Marr. *Vision.* Freeman, San Francisco, CA, 1982.

[128] H. Maturana. Ontology of observing. the biological foundations of self consciousness and the physical domain of existence. In *Conference Workbook: Texts in Cybernetics*, volume 1. American Society For Cybernetics Conference, Felton, CA, American Society For Cybernetics, 1988.

[129] J.L. McClelland and D.E. Rumelhart. An interactive activation model of context effects in letter perception, part i: An account of basic findings. *Psychological Review*, 88:375–407, 1981.

[130] M. McCloskey. Networks and theories: The place of connectionism in cognitive science. *Psychological Science*, 2/6:387–395, 1991.

[131] W.S. McCulloch and W. Pitts. A logical calculus of ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, 5:115–133, 1943.

[132] L.R. Medsker. *Hybrid Intelligent Systems.* Kluwer Academic Publishers, Boston, 1995.

[133] L. Meeden. *Towards planning: incremental investigations into adaptive robot control.* PhD thesis, Indiana University, Bloomington, 1994. http://www.cs.swarthmore.edu/~ meeden/.

[134] M.L. Minsky. *Computation: Finite and Infinite Machines*, pages 32–66. Prentice-Hall, Inc., Englewood Cliffs, NJ, 1967. Chapter 3: Neural Networks. Automata Made up of Parts.

[135] M. Mitchell. *Analogy-Making as Perception: A Computer Model.* MIT Press, Cambridge, MA, 1993.

[136] C. Moore. Dynamical recognizers: Real-time language recognition by analog computers. TR No. 96-05-023, Santa Fe Institute, 1996.

[137] A. Moreno, J. J. Merelo, and A. Etxeberria. Perception, adaptation and learning. In *Proceedings of the Workshop Autopoiesis and Perception*, Dublin, 1992. Dublin City University, August.

[138] C.W. Omlin and C.L. Giles. Constructing deterministic finite-state automata in recurrent neural networks. *Journal of the ACM*, 43(6):937–972, 1996.

[139] M. Page and D. Norris. Modeling immediate serial recall with a localist implementation of the primacy model. In J. Grainger and A.M. Jacobs, editors, *Localist connectionist approaches to human cognition*, volume 1. Erlbaum, Mahwah, NJ, 1998.

[140] J. Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* Morgan Kaufmann, San Francisco, CA, 1988.

[141] N. Pennington and R. Hastie. The story model for juror decision making. In R. Hastie, editor, *Inside the Juror*, volume 1, pages 192–221. Cambridge University Press, New York, 1993.

[142] N. Pennington and R. Hastie. A theory of explanation-based decision making. In G.A. Klein, J. Orasanu, R. Calderwood, and C.E. Zsambok, editors, *Decision making in action: Models and methods*, pages 188–201. Ablex Publishing Corporation, Norwood, NJ, 1993.

[143] D. Perrin. Finite automata. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science, Volume B: Formal Models and Semantics.* The MIT Press, Cambridge, MA, 1990.

[144] W. Phillips and W. Singer. In search of common foundations for cortical computation. *Behavioral and Brain Sciences*, 20:657–722, 1997.

[145] T. A. Plate. Holographic reduced representations. *IEEE Transactions on Neural Networks*, 6(3):623–641, 1995.

[146] T. A. Plate. Analogy retrieval and processing with distributed representations. In Keith Holyoak, Dedre Gentner, and Boicho Kokinov, editors, *Advances in Analogy Research: Integration of Theory and Data from the Cognitive, Computational, and Neural Sciences*, pages 154–163. NBU Series in Cognitive Science, New Bulgarian University, Sofia., 1998. Available at http://www.mcs.vuw.ac.nz/~ tap.

[147] T.A. Plate. Holographic reduced representations: convolution algebra for compositional distributed representations. In J. Myopoulos and R. Reiter, editors, *Proceedings of the Twelfth International Joint Conference on Artificial Intelligence*, pages 30–35. Morgan Kaufmann, 1991.

[148] T.A. Plate. *Distributed Representations and Nested Compositional Structure.* PhD thesis, Department of Computer Science, University of Toronto, 1994. Available from http://www.cs.utoronto.ca/~ tap.

[149] T.A. Plate. Analogy retrieval and processing with distributed vector representations. Technical Report CS-TR-98-4, School of Mathematical and Computer Sciences, Victoria University of Wellington, Wellington, New Zealand., 1998.

[150] T.A. Plate. Structured operations with distributed vector representations. In *In K. Holyoak, D. Gentner, & B. Kokinov (Eds.) Advances in analogy research: Integration of theory and data from the cognitive, computational, and neural sciences*, pages 154–163, Sofia, Bulgaria, 1998. New Bulgarian University. Available from http://www.mcs.vuw.ac.nz/~ tap.

[151] J. B. Pollack. *On Connectionist Models of Natural Language Processing*. PhD thesis, Computer Science Department, University of Illinois, Urbana, 1987. Available as TR MCCS-87-100, Computing Research Laboratory, New Mexico State University, Las Cruces, NM.

[152] J. B. Pollack. The induction of dynamical recognizers. *Machine Learning*, 7:227–252, 1991.

[153] J.B. Pollack. Recursive distributed representations. *Artificial Intelligence*, 46(1-2):77–105, 1990.

[154] S. Quartz and T.J. Sejnowski. The neural basis of cognitive development: A constructivist manifesto. *Behavioral and Brain Sciences*, 20(4):537–596, 1997.

[155] S. Ramanathan and P.V. Rangan. Adaptive feedback techniques for synchronized multimedia retrieval over integrated networks. *IEEE/ACM Transactions on Networking*, 1(2):246–260, 1993.

[156] G.N. Reeke and G.M. Edelman. Real brains and Artificial Intelligence. *Daedalus (Proc. Am. Acad. Arts and Sciences)*, 117(1):143–173, Winter 1988.

[157] G.N. Reeke and G.M. Edelman. A Darwinist view of the prospects for conscious artifacts. In G. Trautteur, editor, *Consciousness: Distinction and Reflection*, chapter 7, pages 106–130. Bibliopolis, Naples, 1995.

[158] T. Regier. *The Human Semantic Potential: Spatial Language and Constrained Connectionism.* MIT Press, Cambridge, MA, 1996.

[159] R.G. Reilly and N.E. Sharkey. *Connectionist Approaches to Natural Language Processing.* Lawrence Erlbaum Associates, Hillsdale, NJ, 1992.

[160] P. Rodriguez, J. Wiles, and J. Elman. How a recurrent neural network learns to count. *Connection Science*, ta.

[161] D. Roth. A connectionist framework for reasoning: Reasoning with examples. In *Proc. of the National Conference on Artificial Intelligence*, pages 1256–1261, 1996.

[162] D. Roth. Learning to resolve natural language ambiguities: A unified approach. In *Proc. of the National Conference on Artificial Intelligence*, pages 806–813, 1998.

[163] D. Roth and D. Zelenko. Part of speech tagging using a network of linear separators. In *COLING-ACL 98, The 17th International Conference on Computational Linguistics*, pages 1136–1142, 1998.

[164] G. Scheler. Pattern classification with adaptive distance measures. Technical Report FKI-188-94, Technische Universitat Munchen, 1994.

[165] L. Shastri. Neurally motivated constraints on the working memory capacity of a production system for parallel processing. In *Proceedings of the Fourteenth Conference of the Cognitive Science Society*, pages 159–164, 1992.

[166] L. Shastri. Structured connectionist model. In M. Arbib, editor, *The Handbook of Brain Theory and Neural Networks*, volume 1, pages 949–952. MIT Press, Cambridge, MA, 1995.

[167] L. Shastri. Temporal synchrony, dynamic bindings, and SHRUTI – a representational but non-classical model of reflexive reasoning. *Behavioral and Brain Sciences*, 19(2):331–337, 1996.

[168] L. Shastri. Advances in SHRUTI — a neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony, 1998. Submitted. Available at http://www.icsi.berkeley/˜ shastri/psfiles/shruti_adv_98.ps.

[169] L. Shastri. Types and quantifiers in SHRUTI — a connectioninst model of rapid reasoning and relational processing, To appear. NIPS-98 Workshop on Hybrid Neural Symbolic Integration.

[170] L. Shastri and V. Ajjanagadde. An optimally efficient limited inference system. In *Proceedings of AAAI-90*, pages 563–570. Boston, MA, 1990.

[171] L. Shastri and V. Ajjanagadde. From simple associations to systematic reasoning: a connectionist representation of rules, variables, and dynamic bindings using temporal synchrony. *Behavioral and Brain Sciences*, 16(3):417–451, 1993.

[172] L. Shastri and D.J. Grannes. A connectionist treatment of negation and inconsistency. In *Proceedings of the Eighteenth Conference of the Cognitive Science Society*, —, 1996. San Diego, CA, —.

[173] L. Shastri, D.J. Grannes, S. Narayanan, and J.A. Feldman. A connectionist encoding of schemas and reactive plans, 1998. Submitted. Available at http://www.icsi.berkeley/~ shastri/psfiles/ulm.ps.

[174] H. Siegelmann. The simple dynamics of super Turing theories. *Theoretical Computer Science*, 168:461–472, 1996.

[175] H. T. Siegelmann and E. D. Sontag. Turing computability with neural nets. *Applied Mathematics Letters*, 4(6):77–80, 1991.

[176] H.T. Siegelmann, B.G. Horne, and C.L. Giles. Computational capabilities of recurrent NARX neural networks. *IEEE Trans. on Systems, Man and Cybernetics - Part B*, 27(2):208, 1997.

[177] H.T. Siegelmann and E.D. Sontag. On the computational power of neural nets. *Journal of Computer and System Sciences*, 50(1):132–150, 1995.

[178] W. Singer. Synchronization of cortical activity and its putative role in information processing and learning. *Ann. Rev. Physiol*, 55(–):349–74, 1993.

[179] P. Smolensky. Tensor product variable binding and the representation of symbolic structures in connectionist systems. *Artificial Intelligence*, 46(1-2):159–216, 1990.

[180] J. Sougne. Simulation of the negative conclusion bias with INFERNET. Technical report, FAPSE-TR 98-001, University of Lige, 1988.

[181] J. Sougne. Solving problems requiring more than 2 instantiations with INFERNET. Technical report, FAPSE-TR 98-002, University of Lige, 1988.

[182] J. Sougne. A connectionist model of reflective reasoning using temporal properties of node firing. In *Proceedings of the Eighteenth Annual Conference of the Cognitive Science Society*, pages 666–671, Mahwah, NJ, 1996. Lawrence Erlbaum Associates.

[183] J. Sougne. Connectionism and the problem of multiple instantiation. *Trends in Cognitive Sciences*, 2(5):183–189, 1998.

[184] J. Sougne. Period doubling as a means of representing multiply instantiated entities. In *Proceedings of the Twentieth Annual Conference of the Cognitive Science Society*, pages 1007–1012, Mahwah, NJ, 1998. Lawrence Erlbaum Associates.

[185] J. Sougne and R.M. French. A neurobiologically inspired model of working memory based on neuronal synchrony and rythmicity. In *Proceedings of the Fourth Neural Computation and Psychology Workshop: Connectionist Representations*, pages 155–167, London, 1997. Springer-Verlag.

[186] A. Sperduti. Labeling RAAM. *Connection Science*, 6(4):429–459, 1994.

[187] A. Sperduti. On the computational power of neural networks for structures. *Neural Networks*, 10(3):395, 1997.

[188] R. Sun. The discrete neuronal model and the probabilistic discrete neuronal model. In *Proc.of International Neural Network Conference, Paris 1990*, pages 902–907, Netherlands, 1990. Kluwer.

[189] R. Sun. The discrete neuronal model and the probabilistic discrete neuronal model. In B.Soucek, editor, *Neural and Intelligent Systems Integration*, volume 1, pages 161–178. John Wiley and Sons, New York, NY, 1991.

[190] R. Sun. On variable binding in connectionist networks. *Connection Science*, 4(2):93–124, 1992.

[191] R. Sun. Robust reasoning: integrating rule-based and similarity-based reasoning. *Artificial Intelligence*, 75(2):241–296, 1995.

[192] R. Sun and F. Alexandre. *Connectionist Symbolic Integration*. Lawrence Erlbaum Associates, Hillsdale, NJ, 1997.

[193] R. Sun and L. Bookman. *Architectures incorporating neural and symbolic processes*. Kluwer, Netherlands, 1994.

[194] R. Sun, E. Merrill, and T. Peterson. A bottom-up model of skill learning. In *Proc. of 20th Cognitive Science Society Conference*, pages 1037–1042, Mahwah, NJ., 1998. Lawrence Erlbaum Associates.

[195] R. Sun and T. Peterson. A subsymbolic+symbolic model for learning sequential navigation. In *Proc. of the Fifth International Conference of Simulation of Adaptive Behavior (SAB'98)*, Cambridge, MA, 1998. MIT Press.

[196] W. Tabor. Dynamical automata. 47 pages. Tech Report # 98-1694. Department of Computer Science, Cornell University. Download from http://cs-tr.cs.cornell.edu/, 1998.

[197] W. Tabor. Context free grammar representation in neural networks. 7 pages. Draft version available at http://simon.cs.cornell.edu/home/tabor/papers.html, submitted to NIPS.

[198] B.B. Thompson, M.S. Cohen, and J.T. Freeman. Metacognitive behavior in adaptive agents. In *Proceedings of the World Congress on Neural Networks*, pages 266–273, Hillsdale, NJ, 1995. IEEE, Lawrence Erlbaum.

[199] S.E. Toulmin. *The Uses of Argument*. Cambridge University Press, Cambridge, 1958.

[200] D.S. Touretzky and G.E. Hinton. A distributed connectionist production system. *Cognitive Science*, 12(3):423–466, 1988.

[201] G. Towell and J. Shavlik. Extracting refined rules from knowledge-based neural networks. *Machine Learning*, 13(1):71–101, 1993.

[202] M. Usher and N. Donnelly. Visual synchrony affects binding and segmentation in perception. *Nature*, 394(-):179–182, 1998.

[203] L. G. Valiant. *Circuits of the Mind*. Oxford University Press, 1994.

[204] L. G. Valiant. A neuroidal architecture for cognitive computation. In *ICALP'98, The International Colloquium on Automata, Languages, and Programming*, pages 642–669. Springer-Verlag, 1998.

[205] C. von der Malsburg. The correlation theory of brain function. Technical Report 81-2, Max-Planck-Institute for Biophysical Chemistry, Gottingen, 1981.

[206] C. von der Malsburg. Am I thinking assemblies? In G. Palm & A. Aertsen, editor, *Brain Theory*, volume 1. Springer-Verlag, 1986.

[207] C. Watkins. *Learning from Delayed Rewards*. PhD thesis, Cambridge University, 1989.

[208] P. Werbos. Advanced forecasting methods for global crisis warning and models of intelligence. In *General Systems Yearbook*, volume 1, page Appendix B. 1977.

[209] P. Werbos. Neural control and supervised learning: An overview and evaluation. In D.White and D.Sofge, editors, *Handbook of Intelligent Control*, chapter 3, pages 65–89. Van Nostrand Reinhold, NY, 1992.

[210] P. Werbos. *The Roots of Backpropagation: From Ordered Derivatives to Neural Networks and Political Forecasting*. Wiley, NY, 1994.

[211] P. Werbos. Backpropagation: past, present, and future. In *Proceedings of International Conference of Neural Networks*, volume 1, pages 343–353, San Diego, CA, 1998. IEEE, IEEE.

[212] S. Wermter. *Hybrid Connectionist Natural Language Processing*. Chapman and Hall, International Thomson Computer Press, London, UK, 1995.

[213] S. Wermter. Hybrid neural and symbolic language processing. Technical Report ICSI-TR-30-97, University of Sunderland, UK, 1997.

[214] S. Wermter and M. Löchel. Learning dialog act processing. In *Proceedings of the International Conference on Computational Linguistics*, pages 740–745. Copenhagen, Denmark, 1996.

[215] S. Wermter and M. Meurer. Building lexical representations dynamically using artificial neural networks. In *Proceedings of the International Conference of the Cognitive Science Society*, pages 802–807. Stanford, 1997.

[216] S. Wermter, E. Riloff, and G. Scheler. *Connectionist, Statistical and Symbolic Approaches to Learning for Natural Language Processing*. Springer, Berlin, 1996.

[217] S. Wermter and V. Weber. SCREEN: Learning a flat syntactic and semantic spoken language analysis using artificial neural networks. *Journal of Artificial Intelligence Research*, 6(1):35–85, 1997.

[218] J. Wiles and J. Elman. Landscapes in recurrent networks. In Johanna D. Moore and Jill Fain Lehman, editors, *Proceedings of the 17th Annual Cognitive Science Conference*. Lawrence Erlbaum Associates, 1995.

[219] D.H. Younger. Recognition and parsing of context-free languages in time n^3. *Information and Control*, 10(2):189 – 208, (1967).

[220] A. Youssef, H. Abdel-Wahab, and K. Maly. A scalable and robust feedback mechanism for adaptive multimediamulticast systems. Technical report, TR 97 38, Old Dominion University, 1997.