A Connectionist Approach to the Story of Over
Author(s): Catherine L. Harris
Proceedings of the Fifteenth Annual Meeting of the Berkeley Linguistics Society (1989), pp. 126-137

A CONNECTIONIST APPROACH TO THE STORY OF OVER

Catherine L. Harris
University of California, San Diego

Introduction

Linguists working in the framework of cognitive linguistics have recently suggested that connectionist networks may provide a computational formalism well suited for the implementation of their theories (Langacker 1987; Lakoff 1987). The appeal of these networks includes the ability to extract the family resemblance structure inhering in a set of input patterns, to represent both rules and exceptions, and to integrate multiple sources of information in a graded fashion. The current paper explores the matches between cognitive linguistics and connectionism by implementing some aspects of Brugman and Lakoff's analysis of the diverse meanings of the preposition *over* (Brugman 1988; Brugman and Lakoff 1988).

I will briefly describe some of the matches between the cognitive linguistics approach and connectionist capabilities, sketch part of the Brugman and Lakoff work, and then present the connectionist model.

Cognitive Linguistics and Connectionism

Two of the attractions of connectionism for cognitive scientists are its use of spreading activation for satisfying multiple, simultaneous constraints (McClelland and Rumelhart 1981; Hinton 1984; Smolensky 1988), and learning algorithms which can discover the statistical regularities existing in a large corpus of patterns (Rumelhart and McClelland 1986; Elman 1988; St. John and McClelland 1988). To explain these properties and how they arise out of connectionist networks, I will focus on models in which a gradient descent learning procedure (such as the back propagation algorithm of Rumelhart, Hinton & Williams 1986) is used to train a network to associate a set of input patterns with a set of output patterns. The patterns are binary strings which are assigned linguistic values by the implementor. For example, in Rumelhart and McClelland's (1986) model of the acquisition of the past tense of English verbs the input patterns represented phonological strings for the present tense of various English verbs, and the output represented phonological strings corresponding to the past-tense form of these verbs.

A network is an arrangements of units, where units are entities which have an activation value and are capable of sending activation to other units through weighted connections. Typically, a network has an input and an output layer of units, and possibly some intermediate or "hidden" layers. A pattern is presented to a network by giving the units of the input layer activation values corresponding to the patterns to be represented. The activation of each unit of the input layer is propagated forward to activate the units on the next layer. The pattern of activation appearing on the units of the output layer constitutes the output pattern.

During training, a teaching pattern is presented for each input. Learning algorithms such as backpropagation provide a method for modifying the weighted connections between units so that each time a given pattern appears on the input layer,

the desired pattern will appear on the output layer. The computational task of the network can thus be seen as finding a configuration of weights which will allow storage of some (possibly very large or, under some circumstances, infinite) number of input-output pairs.

This type of storage has been called "superpositional" or "distributed" to emphasize the contrast with traditional views of how data can be stored by mechanical devices. In the traditional view, distinct items are stored in distinct memory locations, while in distributed systems items are stored on top of one another, or "superimposed" (Rumelhart and Norman 1986). Two things of particular interest to cognitive linguists are that the superpositioning of multiple patterns can allow the invariances among the patterns to emerge (as in the extraction of a prototype from multiple exemplars) and that the detection of invariances by the network can yield a structure which allows novel patterns to be treated on analogy to previously stored patterns.

The Story of Over

The preposition *over*, like other highly frequent English words, can evoke a range of meanings. Brugman (1988) and Brugman and Lakoff (1988) (hereafter B&L) identified three main ways that *over* can indicate a spatial relationship between a trajector (TR) and a landmark (LM). (In their analysis, non-spatial usages are variations on these three schemas. The connectionist model will be restricted to spatial usages.)

I. The **above** schema: The TR is vertically above, but not touching, the LM, as in example (1) below.
II. The **above-across** schema: The TR is an object moving on a path above, and extending beyond, the boundaries of the LM, as in example (2). Alternately, the TR could be a stationary, 1-dimensional object, as in (3). Example (4) shows that this schema allows contact between the TR and the LM.
III. The **cover** schema: The TR is an object whose 2-dimensional extent covers the LM (extends to the edges of or beyond the LM). In most cases, the TR is construed as being vertically above, and in contact with, the LM (5). The TR does not have to be vertically above the LM, as illustrated by (6).

The three schemas are diagrammed in Figure 1. Two variations on the **above-across** schema are shown.

(1) The helicopter hovers over the city.
(2) The plane flies over the bridge.
(3) The line stretches over the wall.
(4) The plane rolls over the bridge.
(5) The cloth is over the table.
(6) The carpet stretches over the wall.

One way the uses of *over* are related is through shared components. For example, the sentences in (7)-(10) share an *above* component. (Some hypothesized features of the expressions in these sentences are listed in italics.)

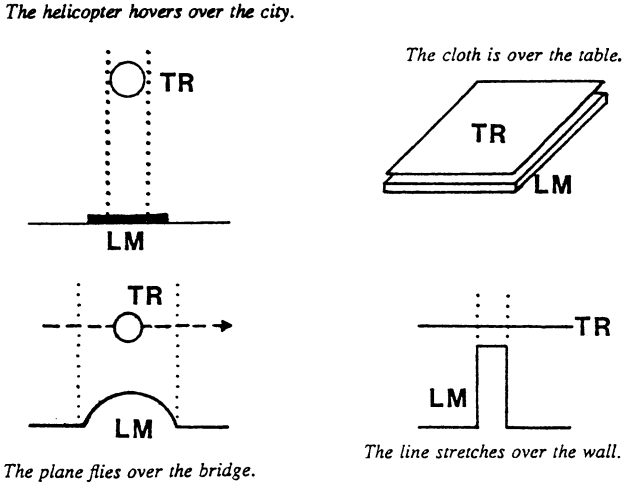

Figure 1. Image schema diagrams for 4 uses of *over*.

(7) The bee is over the table. *above*
(8) The bird flew over the hill. *above across up*
(9) The person walks over the hill. *above across up contact*
(10) The person lives over the hill. *above across up contact end-point*

The component **up** in (8) signals that the trajectory of the bird's flight has an upward component, while the component **contact** in (9) refers to the presence of TR-LM contact. B&L view (10) to be a further variation on (9), in that it shares components with (9) but differs in having a focus on the end-point of the trajectory. (That is, the location of the person's house is specified to be at the end of the path which extends over the hill.)

The traditional approach to representing the different meanings of polysemous words has been to identify word meaning with "core meaning", where "core" is something common to all of a word's multiple uses (Jackendoff 1983). Where core meanings cannot be found, theorists have proposed that polysemes be treated like homonyms: the different meanings must simply be listed separately in the lexical entry of each word. B&L present arguments for why an abstract notion of aboveness could not be the candidate for a core meaning for *over*. They describe how the polysemes of *over* are interrelated and point out that the separate listings approach fails to acknowledge these relationships. In addition, their analysis sheds new light on the problem of how one meaning is selected and integrated with the meanings of other words in the utterance: the meaning evoked by an utterance is the result of constraint satisfaction. An utterance is most felicitous when its component words contribute to a single coherent schema. Successive words in an utterance narrow down or constrain the

number of possible meanings.

Implementation

Connectionist models are good at integrating simultaneous constraints, at extracting prototypes from examples, at representing both rules and exceptions, and at generalizing to new forms on analogy to stored patterns. B&L describe how polysemous words are prototype categories and that the meanings they evoke are the result of constraint satisfaction. Although the mapping from expressions to their meanings cannot be described by a single rule, this mapping does contain a number of regularities. As Rumelhart and McClelland (1986) have shown, connectionist networks are good at representing rules, sub-regularities, and rule-exceptions. For these reasons, the different meanings of *over* appear to be a good domain for exploring the problems and rewards of implementing a cognitive linguistics analysis in a connectionist network.

There are many different ways a model could incorporate the matches between cognitive linguistics and connectionism, and many different methods of implementing aspects of B&L's analysis of *over*. The B&L analysis is rich and complex, and the current model is necessarily a subset. It includes only spatial, non-metaphorical uses of *over*, and only those spatial senses which could be captured with a limited vocabulary and sentence length. Although the model contains many aspects of their analysis, it does not incorporate every distinction made by B&L, and some of their details appear here in slightly altered form. Furthermore, in no way is the model intended to be a "test" of B&L's account of polysemy. Instead, it demonstrates that mechanisms exist which will produce the constructs they hypothesize.

It is difficult to adequately describe the details of the implementation in this brief report. A full account is contained in Harris (1989).

Architecture

The network was given the task of mapping input patterns of the form "trajector verb (over) landmark" to either the **above** schema, the **above-across** schema, or the **cover** schema. Figure 2 shows the number of layers, connections between layers, and the contents of the input and output layers.

The output layer. The output layer consisted of 6 units, one for each of the three schemas, and three to indicate whether the landmark and trajector make contact, whether the trajectory has an upward component, and whether the path has an end-point focus (as is the case with expressions such as *The man lives/is over the hill).* A number of other features could have been included. For example, B&L describe variations on the **above across** schema in which the LM has or doesn't have significant horizontal extent, or in which the TR is 1-dimensional or a multiplex of entities. Although the pattern set (described below) contains TR's and LM's with these properties, such properties were not explicitly represented in the output schema. It was desired that the output schema include only features that emerged when one looked at the level of the schema, not features that would always occur whenever a particular item (e.g. *mountain*) was present in the input sentence.

The input layer. Sentences were limited to four words: "trajector verb (over) landmark." (Because all sentences involved the preposition *over*, no actual unit for *over* was needed.) A vocabulary of trajectors, verbs, and landmarks out of which to construct the sentences was then selected. An attempt to maximize diversity in properties such as dimensionality, size, animacy, and motion was the main criterion for selecting the specific words used in the model.

What method can be used to represent the meanings of these words? B&L found that properties such as the dimensionality of the trajector, vertical height of the landmark, and whether a verb specified TR-LM contact were important in determining which schema an utterance would evoke. One goal of the current model was to see if the network could, like B&L, discover these properties.

Lexical items were represented in a localist fashion: a unique unit of the input layer was used to designate each word. This means that the network received no semantic information about the input words. For example, to present the pattern "man live (over) hill" to the network, the input unit for *man, live,* and *hill* were turned on in the input vector.

In order to use the same set of weighted connections to map a large number of input patterns to their target outputs, I hypothesized (following Hinton 1986) that the network would have to learn, from the distributional regularities in the mapping between TR-verb-LM combinations and their output features, that some input items are similar to others in some context but not in other contexts. For example, some hidden units might learn to respond similarly to *car* and *plane* but differently to *car* and *person.* If the inputs had been given a semantic representation (e.g. *mountain* coded as +tall, +wide) the network would still have to learn what combinations of coded input features could be mapped to which six-feature output vectors. In the current case, however, the analytic task of the network is harder, since all it has is the distribution of mappings from TR-verb-LM triples to the six-feature output vectors.

The architecture of the network is shown in Figure 2.

The Pattern Set

For each trajector, three to nine verbs agreeing with the selectional restrictions of English were selected. The goal was to obtain combinations representative of English sentences, not to list all such patterns exhaustively. These TR-verb combinations plus the plural/singular marker were combined with all possible landmarks to generate 2700 patterns. Not all landmarks made sense with each combination of "trajector plural/singular verb." While balls can rolls over floors and tables, they don't typically roll over oceans. Deleting the most obviously anomalous patterns yielded a final set of 1600 patterns.

Network training

The number of hidden units which appear in Figure 2 was obtained by training the network with increasingly fewer hidden units. The 22 units shown in Figure 2 were the minimum required for the network to learn correctly all 1600 patterns.

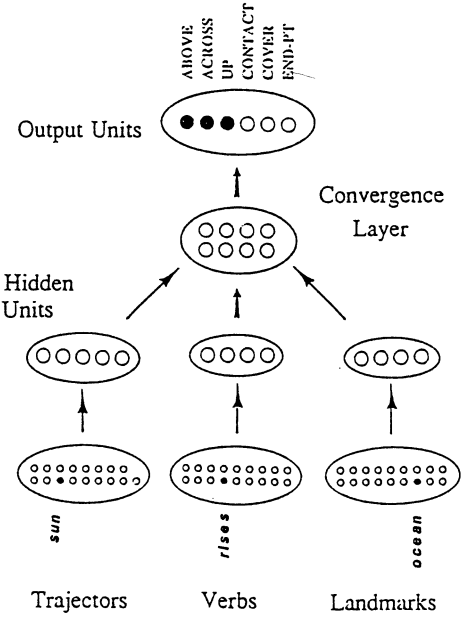

Figure 2. Network architecture.

Weights were modified after every presentation of an input-output pair. Training was considered complete when the network had either achieved correct performance on all patterns, or when the number of correct patterns no longer increased with continued training. The activation of each output unit had to be within 10% of the target unit for a pattern to be considered correct.

The model was run on McClelland and Rumelhart's (1988) BP, a program which implements backward error propagation.

Self-organization of the Hidden Units

Two questions about network behavior are typically asked: (a) how well can the network generalize to patterns on which it has not been trained, and (b) what internal representations has the network constructed in order to learn the input-output mapping. In the current paper, I will focus on the self-organization of the hidden units. Understanding how the hidden units have recoded the input items into abstract categories is a prerequisite for understanding how the network could generalize to novel patterns at all. A complete report of network performance is contained in Harris (1989).

Of the two hidden layers in Figure 2, we are now concerned with only the first layer: the banks of units receiving connections from the input layer. To explore how these units have self-organized during learning, the activation of each hidden unit in response to each of its inputs was recorded. Each hidden unit is separately graphed in Figures 3, 4, and 5 (corresponding to trajector hidden units, verb hidden units, and landmark hidden units).

Visual inspection of the graphs suggests that the hidden units are selectively responding to inputs of a certain type. The inputs which cause the hidden unit to have a high activation value could be called the "receptive field" of the unit. The graphs have been annotated with suggestions about what properties these inputs may have in common. It should be kept in mind that these labels are only hypotheses about the types of recoding the network finds useful in solving the input-output mapping.

Trajectors

Hidden units 1 and 4 appear to be sensitive to the dimensionalities of the trajectors. The two units, however, make different categorizations of the inputs. HU 4 distinguishes between one-dimensional trajectors and all of the other trajectors. The differentiation between these two groups is abrupt compared to the more graded range of values of HU 1. HU 1 roughly divides the trajectors into zero-dimensional, and all non-zero-dimensional. Note that the input unit for "Number" is included in the non-zero group. This is most likely due to the status of the plural marker as a component of a mass entity.

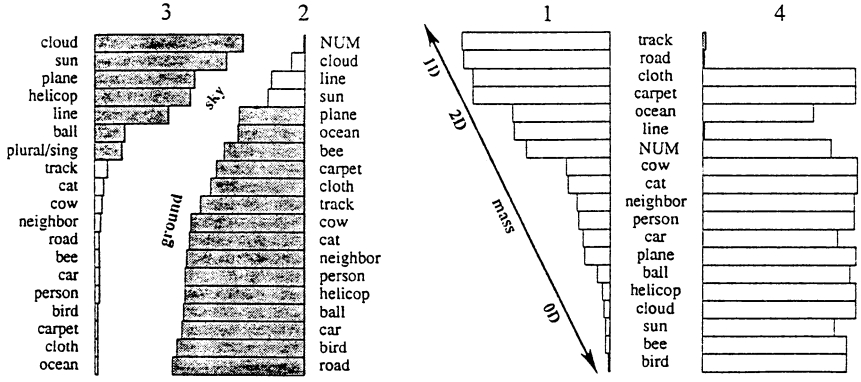

Figure 3: Trajector hidden units.

HU's 2 and 3 grouped their inputs according to whether a trajector was typically a "sky" object (not normally in contact with a surface) or a "ground" object. In HU 3, *plane* and *helicopter* are considered "sky" objects while in HU 2 they are not. This might be because the training set included patterns in which plane and helicopter were in contact with the ground (e.g. when the verb was *roll*).

HU 5 (not depicted) appears to function as a "cloth-carpet" detector. Activations for *cloth* and *carpet* were 1.0, and activations for all other trajectors were under 0.25. These two trajectors participated in **cover** schemas 100% of the time. Because *cloth* and *carpet* were such valid cues to the correct output schema, it was cost-effective for the network to dedicate a hidden unit for detecting their presence in the input.

Verbs

Verb hidden units 2 and 3 distinguished between path verbs and non-path verbs, although they made different divisions. The verbs that allow the **end-point focus** schema (*live, belong, is*) are path verbs in that their schema is **above across up contact endpoint**. Hidden unit 3 groups these with all the other path verbs (*walk, run, fly, lie, rise, roll*). Hidden unit 2, on the other hand, categorizes the **end-point** verbs in the non-path group.

One way to interpret the groupings in hidden units 4 and 1 is as affording information about whether the schema specified by hidden units 2 and 3 should be augmented by the **contact** or **up** features. Hidden unit 4 appears to represent the probability of TR-LM contact, while 1 signals that the verb typically maps to a schema with an **up** feature.

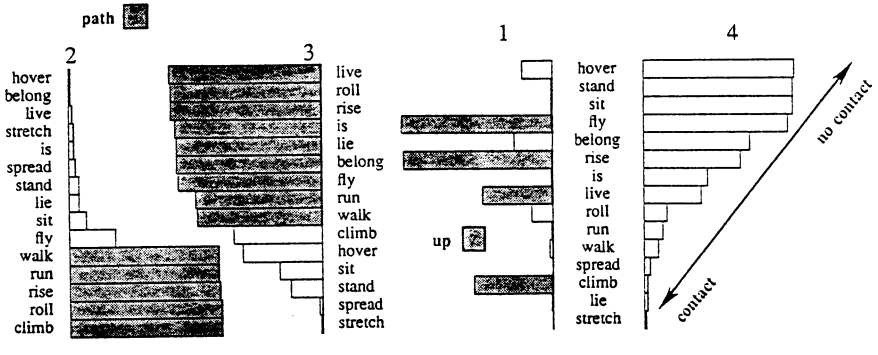

Figure 4: Verb hidden units.

Landmarks

Landmark hidden units 1 and 4 both scale inputs for degree of verticality, but the scales are slightly different. In HU 4, *mountain, hill,* and *bridge* are placed in one group, and *wall, building,* and *house* are in the next-tallest group. In HU 1, however, *bridge* is classified as similar to *wall, building* and *house*. Because both hidden units do turn on in response to the tallest landmarks, we can guess that the network has chosen to encode tall landmarks as the default case.

HU 2 appears to encode the distinction between surfaces and non-surfaces, an important one for predicting whether a path over which mass entities (cats, cows) can spread or lie. The specialization of HU 3 is less clear.

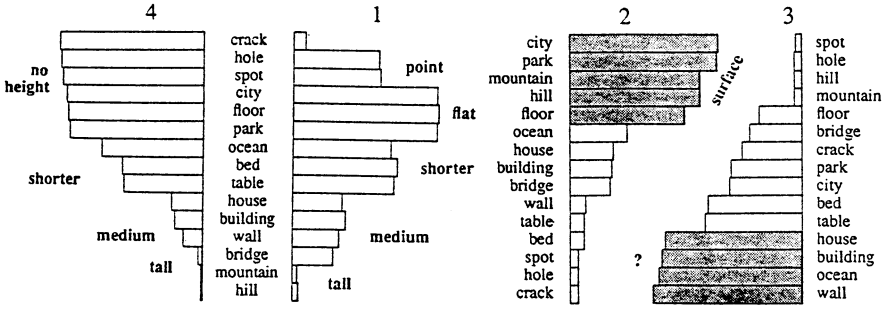

Figure 5. Landmark hidden units.

Extracting Rules from Statistical Regularities

The limited number of hidden units in the current model created an information bottleneck. This bottleneck forced the network to recode the inputs in a highly abstract manner: specific information about the identity of a particular input was mapped into information about salient properties of the input.

These abstract properties could be viewed as the conditional components of rules used by the network to activate the output pattern. For example, the network has extracted two features to characterize the input *person*. From Figure 4 these are "ground TR" and "not-1D TR." (Trajector hidden unit 1 has only a low activation, unit 2 signals "not-1D TR," unit 3 also has only a low activation, unit 4 signals "ground TR," and unit 5 has a low activation). The input pattern "person walk (over) hill" could be understood to activate the following rule:

> IF ground TR, AND (not 1D TR), AND path type1, AND path type2,
> AND surface LM, THEN **above across up contact.**

The hidden unit activations of eight input patterns have been reproduced in Figure 6. I have labeled each of the role-specific hidden units according to what information it appears to be passing on to the next layer (as indicated by the charts in Figures 3 - 5). The activations of the convergence layer have also been included.

If the properties of the role-specific hidden layer are viewed as the abstract components of rules, then the network could, in theory, represent 1600 rules (where a rule is understood to be an action that applies when some preconditions are met). This would be the case if even very related patterns resulted in subtly different hidden unit activations. In the current network, a number of the inputs function as synonyms (compare the activation values for the pairs *cat* and *cow*, *person* and *neighbor*, *run* and *walk*, *hill* and *mountain*, *hole* and *spot* in Figures 3-5). This means that there will be less than 1600 different patterns of activation across the role-specific hidden units. However, it is clear that in theory the number of "rules" could approach the number of patterns seen by the network. This is a desirable feature when the regularities to be

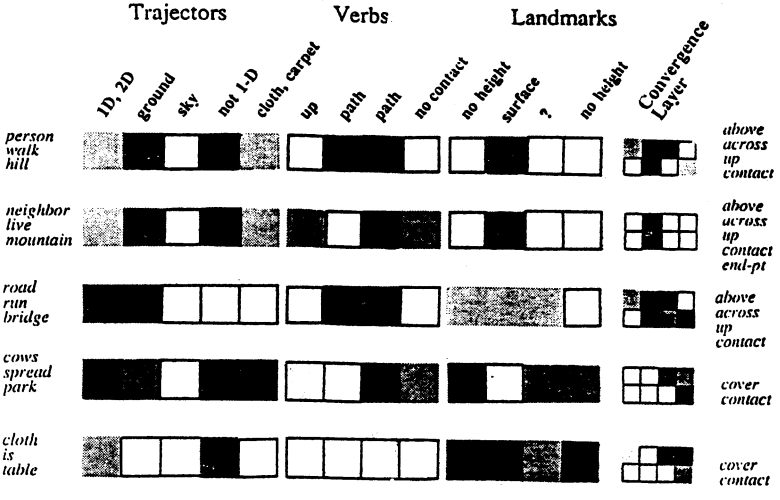

Figure 6. Hidden unit activations for five input-output pairs.

extracted contain complex sub-patterns and are dependent on contextual variation, or when a system must be continually accommodating old forms to the new forms of a changing environment.

Analogy

A system can be understood to represent an analogy between two different patterns if it has a means of recoding them into patterns which are identical or (by some criterion) sufficiently similar.

Figures 3-6 show that the network did develop such a system: the hidden layer functions to map specific inputs such as *car run hill* into abstract properties like "0D trajector" and "tall LM." These abstract properties could then be used to interpret novel patterns. For example, although the network had never been given a *car fly* pattern, it has learned that cars are similar to planes and helicopters (see trajector hidden unit 1, 4, and 2, Figure 4). Given this similarity, it is not surprising that the network responded to the novel pattern, "car fly (over) LM" on analogy to the learned pattern "plane/helicopter fly (over) LM."

More difficult for the network was the novel pattern "carpet fly (over) LM". This pattern was difficult because it contains a conflict of schemas. All examples involving *carpet* that the network was given had **cover** schemas as their target output. In contrast, the verb *fly* always activated the **above-across** schema. Because the network was never given any examples in which it had to resolve the conflict, it didn't know whether the rule for *fly* outweighed the rule for *carpet*, and so activated both schemas simultaneously.

Discussion

The connectionist research program outlined by McClelland and Rumelhart (1986) and Smolensky (1988) looks for correspondences between the representational and processing capacities of connectionist systems and general cognitive phenomena. The correspondences between cognitive linguistics and connectionism are intriguing. B&L's account of polysemy points to the need for a mechanism that can induce categories from a set of examples, learn to extract rules from rule-governed data, and resolve conflicts in rules by constraint satisfaction. The model described here contributes to past work (e.g., Anderson 1983; Elman 1988; Rumelhart & McClelland 1986; Hinton 1986) in showing how networks provide such a mechanism.

Although the model fares well in illustrating how connectionist networks extract the statistical regularities of input data and construct internal representations which support analogy, its limitations as a model of the polysemies of *over* are sobering. Few of the senses of *over* are captured, only a single lexical item is represented, and the schema transformations posited by B&L are not included.

The method used for representing combinations of words (the input units) and the meanings of expressions (the output units) is an awkward one. The network was given no information about what individual lexical items mean. Instead, it received information about the meaning of whole expressions. This strategy was adopted to ensure that the problem of mapping sentences to their meanings was not made too trivial. For current purposes, this was a wise choice, since one of the successes of the model was that, under pressure to solve the mapping task, it constructed a sophisticated system for recoding the inputs into their abstract properties. Nevertheless, the model would be more intuitively pleasing if lexical items were given some of the semantic richness which characterizes our conceptualization of words like *road, plane, fly,* etc.

The meaning of *over* was defined to be various combinations of the elements in the set **above across up cover contact end-point.** These elements were supposed to be a short hand for the image schemas evoked upon encountering an *over* expression. Ideally, however, the schema would not be something given or taught to the network, but something the network constructs in the effort to make sense of the mapping from linguistic units to some very rich internal conceptualization.

At present, connectionist networks provide metaphors for understanding how categories might be induced from examples and how rules, regularities and exceptions could be learned and processed by a single mechanism. Whether these networks can be useful computational tools for linguists will depend on whether small successes like the current work can be repeated on a larger, more complicated scale.

Anderson, J. A. (1983). Cognitive and psychological computation with neural models. *IEEE Transactions on Systems, Man and Cybernetics, 13,* 799-815.

Brugman, C. (1981). *The story of 'over.'* M.A. Thesis, University of California at Berkeley. Available from the Indiana University Linguistics Club.

Brugman, C. (1988). *The story of 'over': Polysemy, semantics and the structure of the lexicon.* Garland Press.

Brugman, C. and Lakoff, G. (1988). Cognitive topology and lexical networks. In G.

W. Cottrell, S. Small, and M. K. Tannenhause (Eds.), *Lexical ambiguity resolution: perspectives from psycholinguistics, neuropsychology and artificial intelligence.* San Mateo, CA: Morgan Kaufman Publishers.

Elman, J. (1988). Finding structure in time. CRL Technical report 8801, Center for Research in Language, University of California, San Diego.

Harris, C. L. (1989). Connectionist explorations in cognitive linguistics. Submitted manuscript. Department of Cognitive Science, University of California, San Diego.

Hinton, G. (1984). Parallel computations for controlling an arm. *Journal of Motor Behavior. 16,* 171-194.

Hinton, G. (1986). Learning distributed representations of concepts. *Proceedings of the 8th annual cognitive science society conference.* Hillsdale, New Jersey: Lawrence Erlbaum.

Jackendoff, R. (1983). *Semantics and cognition.* Cambridge, MA: MIT Press.

Lakoff, G. (1987). Connectionism and cognitive semantics. Seminar presented at University of California, San Diego, spring 1987.

Langacker, R. W. (1987). The cognitive perspective. *CRL Newsletter Vol. 1, No. 3.* Center for Research in Language, University of California, San Diego.

McClelland, J. L., & Rumelhart, D. E. (1981). An interactive activation model of context effects in letter perception: Part I. An account of basic findings. *Psychological Review, 88,* 375-407.

McClelland, J. L., & Rumelhart, D. E. (1986). The appeal of parallel distributed processing. In D. E. Rumelhart and J. L. McClelland (Eds.) *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 1* Cambridge, Mass.: MIT Press.

McClelland, J. L., & Rumelhart, D. E. (1988). *Exploring parallel distributed processing: A handbook of models and programs.* Cambridge, Mass.: MIT Press.

Rumelhart, E. E., Hinton, G., & Williams, R. (1986). Learning internal representations by error propagation. In D. E. Rumelhart and J. L. McClelland (Eds.) *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 1* Cambridge, Mass.: MIT Press.

Rumelhart, D. E., & McClelland, J. L. (1986). Learning the past tense of English verbs. In J. L. McClelland and D. E. Rumelhart (Eds.) *Parallel distributed processing: Explorations in the microstructure of cognition, Vol. 2* Cambridge, Mass.: MIT Press.

Rumelhart, D. E., & Norman, D. A. (1986). Representation in memory. In Stanley Stevens, (Ed.) *Handbook of Experimental Psychology.* New York: Wiley.

Smolensky, P. (1988). On the proper treatment of connectionism. In *Behavior and Brain Sciences, 11,* 1-74.

St. John, M. F., & McClelland, J. L. (1988). Learning and applying contextual constraints in sentence comprehension. *Proceedings of the 10th annual cognitive science society conference.* Hillsdale, New Jersey: Lawrence Erlbaum.