

Appendix 2: Knowledge Representation Analysis

Thomas R. Shultz and Alan C. Bale

This appendix contains knowledge representation analyses of networks from Shultz and Bale's paper *Neural Network Simulation of Infant Familiarization to Artificial Sentences: Rule-like Behavior Without Explicit Rules and Variables*. Three different analysis techniques are employed, focusing respectively on unit activations, connection weights, and contributions, which are products of sending-unit activations and connection weights. Network knowledge representations were assessed at various key points in learning by recording connection weights and network contributions at the end of each output phase. Hidden unit activations were recorded at the end of training only.

Connection Weights

Connection weight diagrams for a single representative network (Network 1 from the ABA condition of Experiment 1) are shown in Figures 1 and 2. Figure 1 shows the network's weights at the end of the second output phase, when it has fully adjusted to the first hidden unit. Figure 2 shows the same network after training has been completed, at which time this network has two hidden units.

In each diagram, the connection weights entering a receiving unit are contained within a rectangular band placed just to the right of the index number of the receiving unit. Inside of each rectangular band, weights are labeled by the sending unit and are portrayed by the color and size of several squares. White squares indicate positive (excitatory) weights, and black squares indicate negative (inhibitory) weights. The size of each weight is represented by the relative size of each square. A label of 0 identifies the bias unit¹, labels of 1-6 identify input units, the last six numbers identify output units, and the remaining numbers identify hidden units. Input and output units are ordered as follows: consonant of first word, vowel of first word, consonant of second word, vowel of second word, consonant of third word, and vowel of third word. The letters A, B, and A to the right of output-unit pairs indicate the syntactic category of each word. Representative patterns in the relative sizes of some of these connection weights can be found in each experiment in all of the networks that we have examined.

¹ The bias unit in a network always has an input of 1 and is connected by trainable weights to each downstream (i.e., non-input) unit in the network. Such weights from a bias unit essentially specify the resting activity threshold of each downstream unit.

In Figure 1, which portrays the network with one hidden unit, the patterns of output weights for the two A-category words, represented by units 8-9 and 12-13, are highly similar, reflecting the common category of these two words. The relatively large weights to these outputs from hidden unit 7 indicate that the network has learned to recognize the first and third words in the training sentences, using that hidden unit. In contrast, the relatively small weights from hidden unit 7 to output units 10 and 11 suggest that the network is not yet recognizing the B-category words.

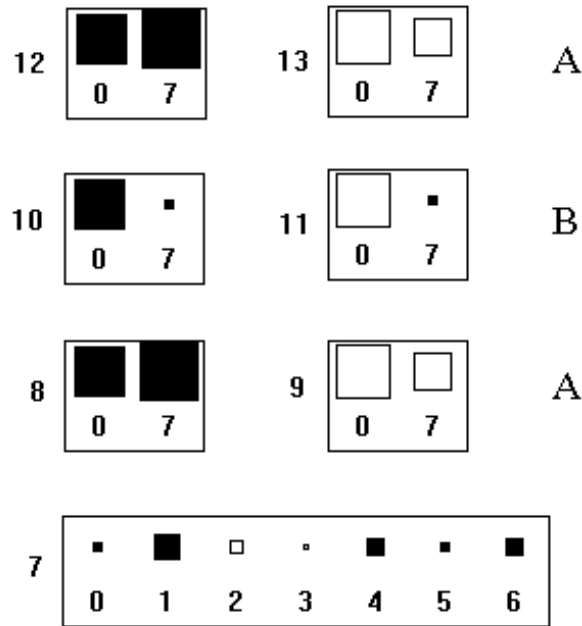


Figure 1. Weights in Network 1 with one hidden unit in the ABA condition of Experiment 1 at the end of the second output phase. Input units are labeled 0-6, the hidden unit is labeled 7, and output units are labeled 8-13. 0 is the bias unit, 1 and 8 represent the consonant of the first word, 2 and 9 represent the vowel of the first word, 3 and 10 represent the consonant of the second word, 4 and 11 represent the vowel of the second word, 5 and 12 represent the consonant of third word, and 6 and 13 represent the vowel of the third word.

Figure 2 shows the same network at the end of training, with two hidden units. The patterns of output weights for the two A-category words, represented by units 9-10 and 13-14, remain highly similar, reflecting the common category of these two words. The relatively large weights to these outputs from hidden unit 7 continue to indicate that hidden unit 7 has the job of recognizing the category of the first and third words. The newer hidden unit 8 has the job of recognizing the category of the second word, as indicated by its relatively large weights to outputs 11 and 12. At this point, the network is accurately recognizing whole sentences in the training set.

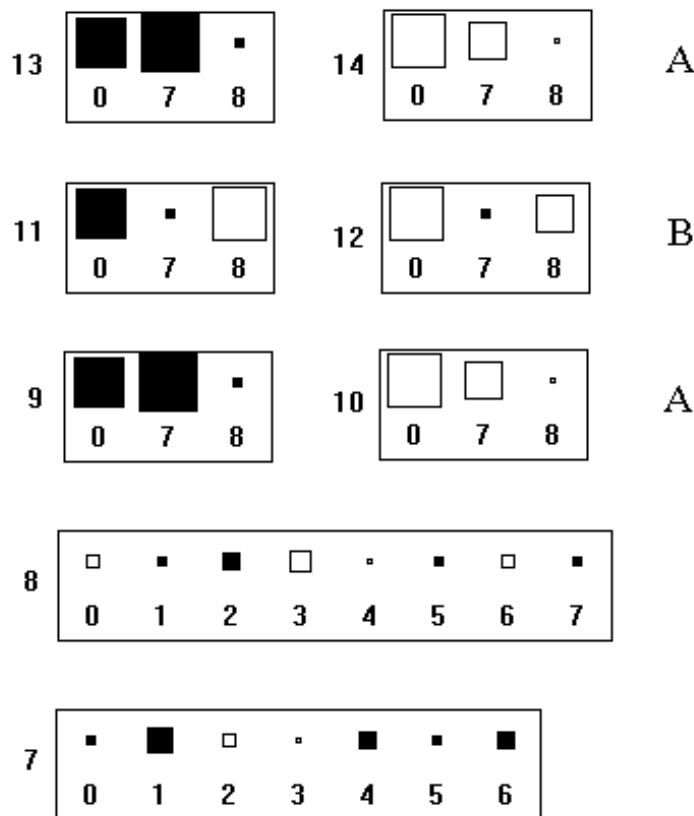


Figure 2. Weights in Network 1 with two hidden units in the ABA condition of Experiment 1 after training has been completed. Input units are labeled 0-6, hidden units are labeled 7-8, and output units are labeled 9-14. 0 is the bias unit, 1 and 9 represent the consonant of the first word, 2 and 10 represent the vowel of the first word, 3 and 11 represent the consonant of the second word, 4 and 12 represent the vowel of the second word, 5 and 13 represent the consonant of third word, and 6 and 14 represent the vowel of the third word.

Figure 3 shows a connection weight diagram for a network with one hidden unit in the ABB condition of Experiment 1. The letters A, B, and B to the right of output-unit pairs again indicate the syntactic category of each word. The patterns of output weights for the two B-category words, represented by units 10-11 and 12-13, are very similar, reflecting the common category of these two words. The relatively large weights to these outputs from hidden unit 7 indicate that the network has learned to recognize the second and third words in the training sentences, using that hidden unit. In contrast, the relatively small weights from hidden unit 7 to output units 8 and 9 suggest that the network is not yet recognizing the A-category words.

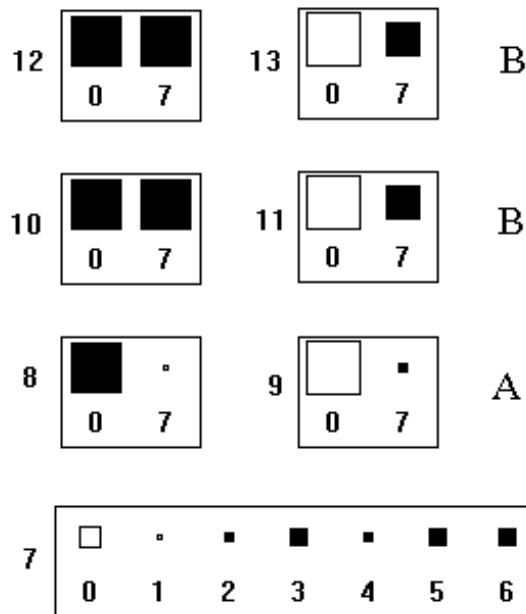


Figure 3. Weights in Network 0 with one hidden unit in the ABB condition of Experiment 1 at the end of the second output phase. Input units are labeled 0-6, the hidden unit is labeled 7, and output units are labeled 8-13. 0 is the bias unit, 1 and 8 represent the consonant of the first word, 2 and 9 represent the vowel of the first word, 3 and 10 represent the consonant of the second word, 4 and 11 represent the vowel of the second word, 5 and 12 represent the consonant of third word, and 6 and 13 represent the vowel of the third word.

Figure 4 shows the weight diagram for this same network at the end of training, when it has two hidden units. The patterns of output weights for the two B-category words, represented by units 11-12 and 13-14, remain very similar, reflecting the common category of these two words. The relatively large weights to these outputs from hidden unit 7 continue to indicate that hidden unit 7 has the job of recognizing the category of the second and third words. The newer hidden unit 8 has the job of recognizing the category of the first word, as indicated by its relatively large weights to outputs 9 and 10. At this point, the network is accurately recognizing whole sentences in the training set.

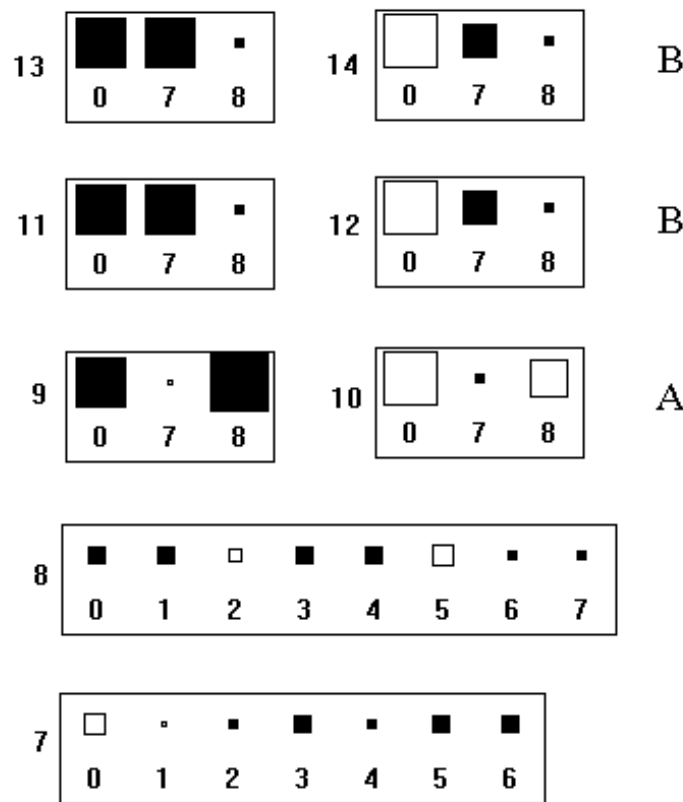


Figure 4. Weights in Network 0 with two hidden units in the ABB condition of Experiment 1 after training has been completed. Input units are labeled 0-6, hidden units are labeled 7-8, and output units are labeled 9-14. 0 is the bias unit, 1 and 9 represent the consonant of the first word, 2 and 10 represent the vowel of the first word, 3 and 11 represent the consonant of the second word, 4 and 12 represent the vowel of the second word, 5 and 13 represent the consonant of third word, and 6 and 14 represent the vowel of the third word.

These connection weight plots show then that networks implement the two duplicate words in sentences (the A word in ABA sentences, and the B word in ABB sentences) with similar sets of weights entering the output units representing the duplicate words. Output weights entering the output units representing the single word (B in ABA sentences, and A in ABB sentences) are distinctly different from the weights for the duplicate words. This pattern of connection weights is replicated across all of the other networks that we have examined in each of the three experiments.

One other representative pattern that the reader may have noticed is that bias weights (from unit 0) entering the output units (identified by the highest six numbers) are strongly negative for units representing consonants and strongly positive for units representing vowels. This reflects that fact that networks learn to expect consonants to have negative sonority values and vowels to have positive sonority values, characteristics that apply to our sonority scale.

Any other patterns in the connection weights of these two networks are not replicated across other networks, as each network implements its own unique solution to the problem of recognizing the experimental sentences. In particular, there appears to be no replicable pattern of connection weights to the hidden units, but as seen later, it is possible to analyze the hidden units in terms of their activation patterns. The connection weight plots suggest that, because large output weights emerge for the duplicated word before the single word, the duplicated words would be mastered first, before the single word.

It is important to realize that these relatively simple solutions of virtually duplicating the weights to outputs representing duplicate words would not work as the task becomes realistically more complex. For example, it would be more typical for grammars to allow multiple word tokens in a particular syntactic category, not just a single word. As specific examples from English, one can say not only *John loves John*, but also *John loves Mary* (two distinct A words in an ABA pattern) and *John hates Mary* (two distinct B words, *loves* and *hates*, in an ABA pattern). Simply duplicating output weights would not suffice in learning to recognize such sentences.

Another realistic complication would involve learning to recognize more than a single sentence type. We trained a few networks, for example, to simultaneously recognize both ABA and ABB sentences from Experiment 1. As this is a more difficult problem, such networks reached victory in between 86 and 149 epochs, with a mean of 113, and recruited from 4 to 7 hidden units, with a mean of 5.5. Connection weights from a small, though otherwise representative, network are diagrammed in Figure 5 at the end of training. Bias weights to the outputs are still negative to units representing consonants (units 11, 13, and 15 in Figure 7) and positive to units representing vowels (units 12, 14, and 16).

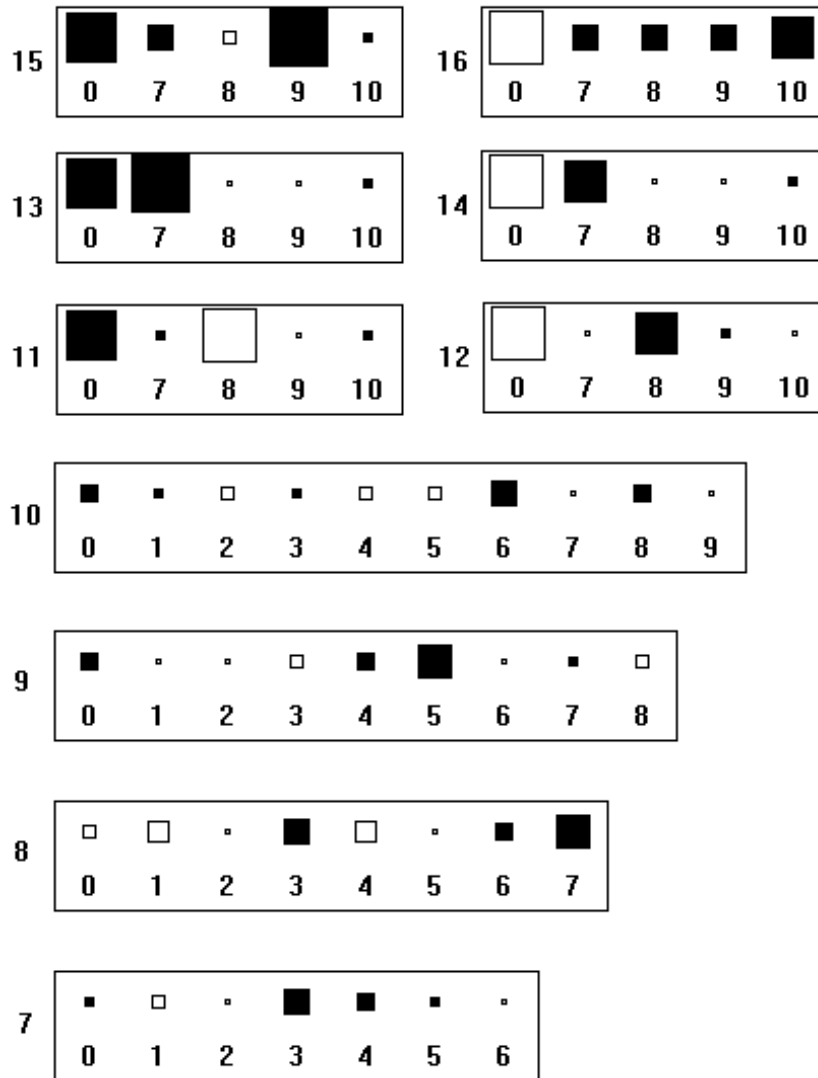


Figure 5. Weights in Network 3 from the ABA&ABB condition using stimuli from Experiment 1, after training, with four hidden units. Input units are labeled 0-6, hidden units are labeled 7-10, and output units are labeled 11-16. 0 is the bias unit, 1 and 11 represent the consonant of the first word, 2 and 12 represent the vowel of the first word, 3 and 13 represent the consonant of the second word, 4 and 14 represent the vowel of the second word, 5 and 15 represent the consonant of third word, and 6 and 16 represent the vowel of the third word.

There is also a tendency for output weights to be resolved for first and second words (units 11-14) before they are resolved for third words (units 15-16). That is, the latest large weights from hidden to output units are typically associated with the third word in the sentence. This makes sense because the category of the third word is most in doubt in these ABA and ABB sentences.

Otherwise, however, there is no discernable pattern in connection weights that replicates across these more complex networks. In particular, a simple duplicating of weights to outputs representing the duplicated word does not suffice, because the duplicated word is different in the two sentence types being learned. When cascade-correlation networks get this complex, a PCA of contributions is often more revealing than are weight diagrams.

In summary, the analyses of connection weights reveal that networks learn to encode the duplicate word before they learn to encode the single word in these three word sentences. The networks learn to decode the two duplicate words using similar sets of weights entering the output units that represent the duplicate words. Weights from the bias unit learned to encode a distinction between consonants (with negative sonority values) and vowels (with positive sonority values).

PCA of Contributions

Contributions are the products of sending-unit activations and connection weights going into output units (Sanger, 1989). Because the net input to a unit is the sum of such products, contributions represent all of the influence on the output units. Sometimes the effects of connection weights can be swamped by large activations, or conversely the effects of unit activations can be swamped by large connection weights. The ability to deal with such imbalances, coupled with the ability to take account of both cross-connections and layer-to-layer connections, make contribution analysis a valuable addition to the network analysis toolkit.

Because a sizeable network has many contributions and there is a distinct set of contributions for each stimulus pattern, complex contribution matrices are often simplified by subjecting them to a Principal Components Analysis (Sanger, 1989). PCA is a data reduction technique for detecting the major independent features of variation in a dataset by taking advantage of the fact that variables are correlated (Jolliffe, 1986). In this case, the variables are network contributions. PCA can provide a compelling picture of knowledge representations in cascade-correlation networks when applied to the covariance matrix of contributions and when the solution is subjected to a varimax rotation in order to improve interpretability of principal components (Shultz, Oshima-Takane, & Takane, 1995).

Contributions from a few networks in each condition of each experiment were recorded at the end of each output phase when networks had fully adjusted to each newly recruited hidden unit (including at the end of training). Each contribution matrix (contributions x input patterns) was converted to covariance form and subjected to PCA with a varimax rotation. Only eigenvalues greater than the mean eigenvalue were retained in the analysis.

At the end of training, the PCA invariably yielded two principal components and plots of component scores resembling that in Figure 6. In the case of this particular network, an ABA-trained network from Experiment 1, the two components accounted for 51.2% and 48.8% of the re-scaled variance in contributions, respectively. The four clusters of the 16 training patterns

shown in Figure 6 reflected mean sonority sums for the A and B categories. Component 1, with a large loading from the second hidden unit, reflected sonority variation in the B category, while Component 2, with a large loading from the first hidden unit, reflected sonority variation in the A category. As can be seen in Figure 6, this is an elegant solution in which sentences with small vs. large sonority sums for the A and B categories are separated in a nearly binary fashion.²

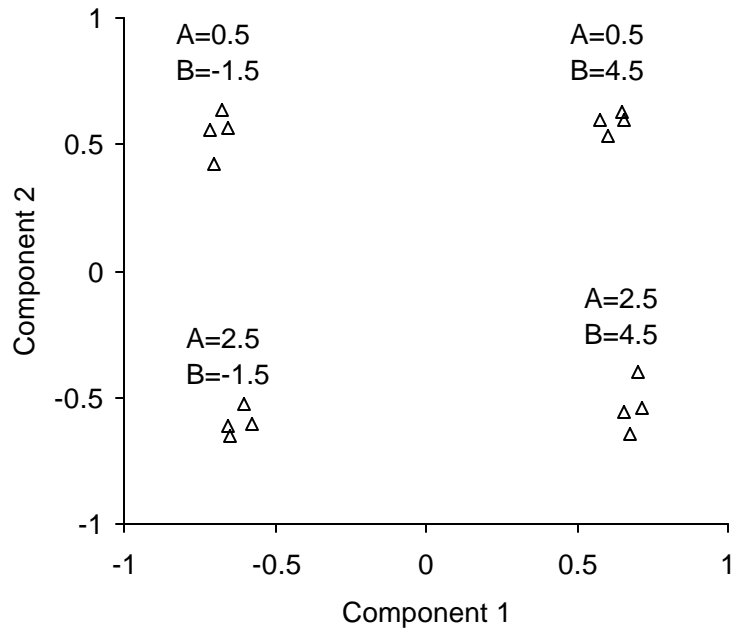


Figure 6. Component scores on the 16 training patterns for a network after training on ABA sentences, along with mean sonority sums for the A- and B-category words in each sentence. Sonority variation of A-category (Component 2) and B-category (Component 1) words is represented.

Notice that the components do not reveal how sentences are categorized into the ABA pattern. This is because PCA collapses correlated contributions into a smaller number of components. Recall that there were very similar connection weights to output units representing duplicate words. Because these similar weights ensure highly correlated contributions across the 16 training patterns, they are collapsed into a single principal component.

² These components can equally well be labeled by sonority differences as well as by sonority sums. The same is true of all subsequent PCAs.

Earlier with only one hidden unit, as shown in Figure 7, this network had a single-component solution emphasizing sonority variation in only the category-A words. As in the foregoing analysis of connection weight diagrams, this makes sense because this network was familiarized to ABA patterns, where category-A words would naturally receive more attention because they are twice as frequent. Here the single component accounts for 100% of the re-scaled variance in network contributions.

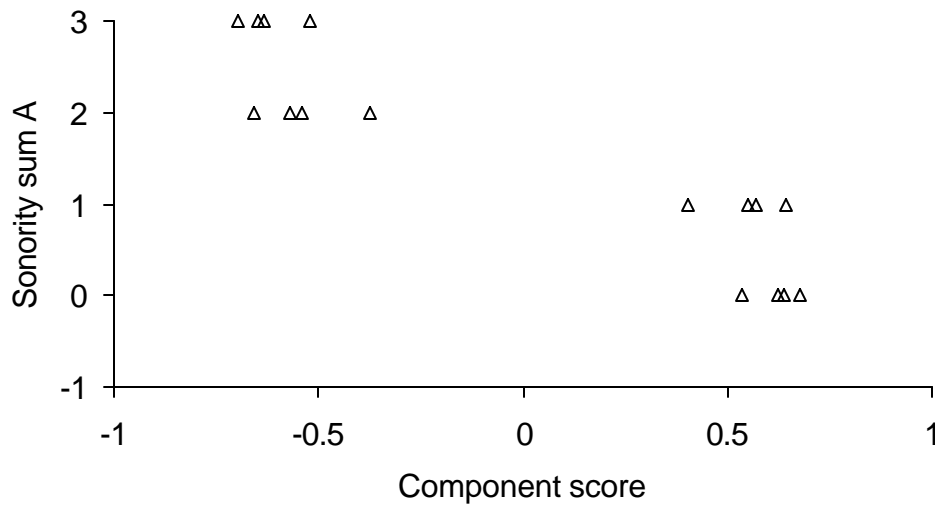


Figure 7. Mean sonority sums in A-category words as a function of component scores for the 16 ABA training patterns in a network with a single hidden unit at the end of the second output phase. The single component represents sonority variation of the duplicated (A) word.

We conducted similar PCAs of contributions on networks that were simultaneously learning both the ABA and ABB sentence patterns. Component score plots for one such network are shown in Figures 8 and 9, after two and four hidden units, respectively. After two hidden units, the PCA yields two components accounting for 51.5% and 48.4% of the re-scaled variance in network contributions. A plot of the component scores for the 16 ABA and 16 ABB training patterns is shown in Figure 8, along with mean sonority sums for each of the four clusters of component scores. Component 1 represents sonority variation in the A category, while Component 2 represents sonority variation in the B category. That this is insufficient for distinguishing ABA from ABB sentences is indicated by the fact that each cluster contains patterns from each of the two sentence types.

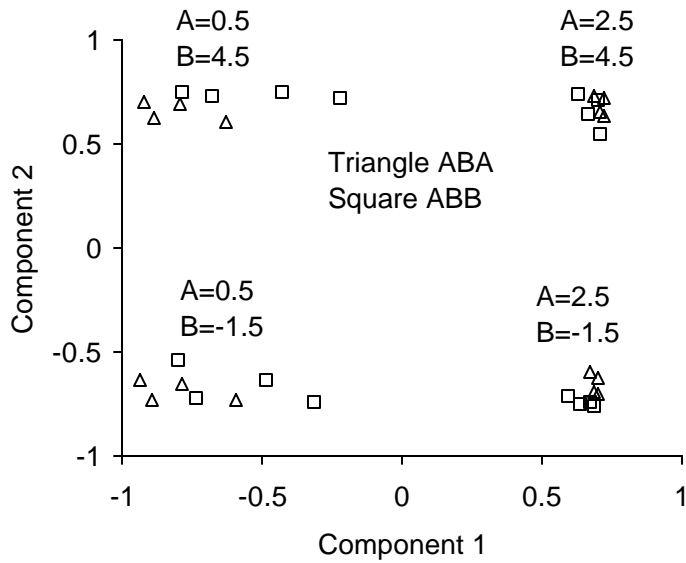


Figure 8. Component scores on the 16 ABA and 16 ABB training patterns for a network with two hidden units, at the end of the third output phase, along with mean sonority sums for the A- and B-category words in each sentence.

After training is completed, with a total of four hidden units, separation of the sentence types greatly improves, as shown in Figure 9. Now the PCA yields three components, accounting for 41.2%, 26.0%, and 26.0% of the re-scaled variance in network contributions. As can be seen in Figure 9, the interaction between Components 1 and 2 neatly separates the two sentence types, with ABA sentences clustering in the upper left and lower right quadrants and ABB sentences clustering in the lower left and right upper quadrants. As noted earlier, mere duplication of the output weights for outputs representing duplicate words is insufficient in this two-sentence-type problem. Consequently, the contributions reflecting the sentence-type distinction do not correlate well enough to be collapsed within any of the components, and are thus revealed in the PCA component scores. Although Component 2 can be seen to represent sonority variation in the A-category words, it is not immediately apparent from the mean sonority sums in Figure 9 how variation in the B-category words is achieved.

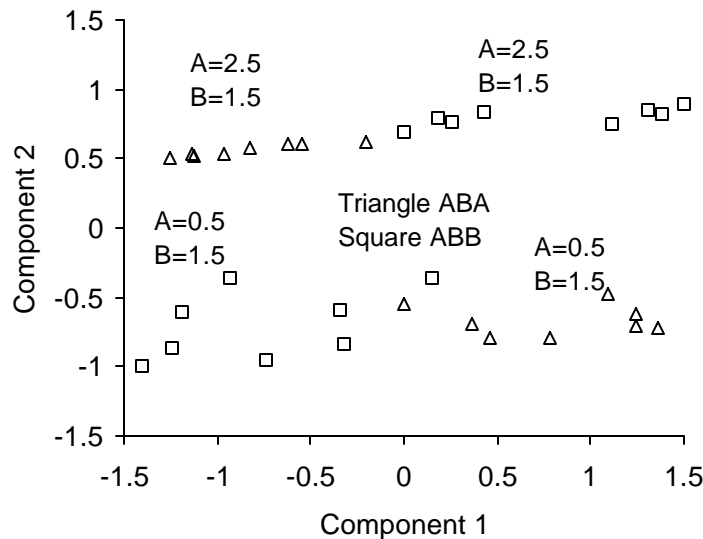


Figure 9. Component scores on the 16 ABA and 16 ABB training patterns for a network after training, along with mean sonority sums for the A- and B-category words in each sentence. The interaction of these two components distinguishes ABA from ABB sentences. Component 2 also represents variation in the sonority sums of A-category words. Some variation in the sonority sums of B-category words (not labeled) is represented by Component 1.

Hidden Unit Activations

The foregoing analyses of weights and contributions paint a clear picture of the impact of hidden units on network outputs. What they do not illuminate is how the hidden units integrate information from the input units and earlier hidden units. The connection weights entering hidden units had fairly obscure patterns (in Figures 1-5), and contributions focused only on the influences on output units. There were suggestions, from both weight diagrams and contribution analysis, that the first hidden unit represented sonority variation in the duplicated word and that the second hidden unit represented sonority variation in the single word. To more directly study how hidden units integrate their inputs, we examined the activation patterns they exhibited on different input patterns. Such activation effectively summarizes the information from both input units and previous hidden units.

We ran a few networks in each condition of each experiment, and recorded hidden unit activation at the end of training. Because input weights to hidden units are frozen after recruitment, there is no need to repeat this analysis during each output phase as was required in the previous analyses. For each network, we plotted the relation between hidden unit activation and sonority sums of the A and B categories in the training patterns. Figure 11 shows two such plots for a representative network in the ABA condition of Experiment 1. The plot on the left of Figure 11 shows a negative relation between activation of Hidden Unit 1 and the sum of sonority values (consonant plus vowel) for the category-A words. The plot on the right of Figure 11 shows a positive relation between activation of Hidden Unit 2 and the sum of sonority values for the category-B words.

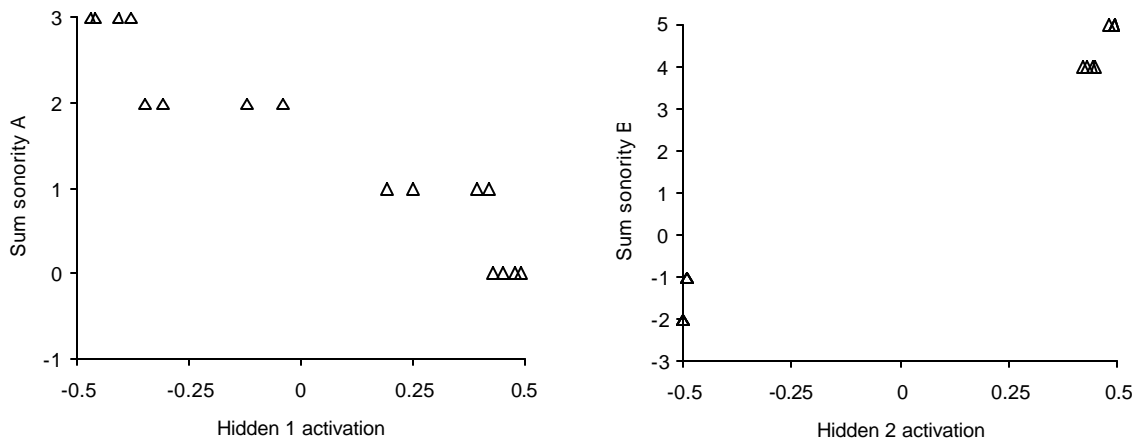


Figure 11. Relation between hidden unit activation and sonority sums in a network at the end of training in the ABA condition of Experiment 1.

Figure 12 shows two similar plots for a representative network in the ABB condition of Experiment 3. The plot on the left of Figure 12 shows a negative relation between activation of Hidden Unit 1 and the sum of sonority values for category-B words. The plot on the right of Figure 12 shows a positive relation between activation of Hidden Unit 2 and the sum of sonority values for category-A words.

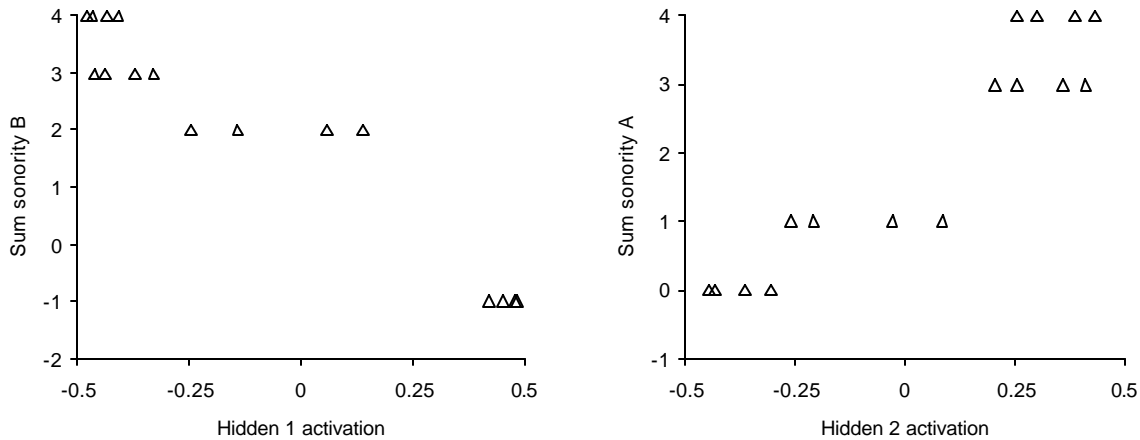


Figure 12. Relation between hidden unit activation and sonority sums in a network at the end of training in the ABB condition of Experiment 3.

As with the other network analysis techniques, we could equally well have substituted sonority differences for sonority sums in these plots. The results and conclusions would be essentially the same. Plots of sonority variation in either consonants or vowels alone also produce much the same result, although the relations with hidden unit activations are not as strong as when using sonority sums or differences.

The generalization that can be drawn from these results is that the first hidden unit represents sonority variation in the duplicate-word category, whereas the second hidden unit represents sonority variation in the single-word category. Again, this is a natural result of networks focusing on the largest current source of error. Because duplicate words initially generate about twice as much total error as do single words, networks deal first with the duplicate-word category. This was true in all of the networks that we have studied. The roles of additional hidden units (beyond two), when recruited, were not so easily identified by examining hidden unit activations. Presumably, the job of these additional hidden units is to clean up any remaining sources of error that the first two hidden units cannot handle. How they accomplish this varies across networks.

Thus, the highly variable pattern of connection weights entering the first two hidden units implements consistent functions to represent sonority variation in the word category that is

generating the largest current source of error. Each network achieves these functions in a distinct way, but the functions themselves appear critical to a successful solution of the grammar-recognition problem. Sonority variation within each word category must be well represented in the hidden unit codes because this variation needs to be reproduced accurately on the output units.

To summarize the hidden unit activation analysis, the first hidden unit learns to encode the sonority sum of the duplicate word, and the second hidden unit learns to encode the sonority sum of the single word. This is a natural consequence of the fact that network learning consistently focuses on reducing as much error as possible.

References

- Jolliffe, I. T. (1986). *Principal components analysis*. Berlin: Springer Verlag.
- Sanger, D. (1989). Contribution analysis: A technique for assigning responsibilities to hidden units in connectionist networks. *Connection Science*, 1, 115-138.
- Shultz, T. R., Oshima-Takane, Y., & Takane, Y. (1995). Analysis of unstandardized contributions in cross connected networks. In D. Touretzky, G. Tesauro, & T. K. Leen, (Eds). *Advances in Neural Information Processing Systems 7* (pp. 601-608). Cambridge, MA: MIT Press.