Segmenting State into Entities and its Implication for Learning

James Henderson

Department of Computer Science, University of Exeter, Exeter EX4 4PT, United Kingdom
`J.B.Henderson@ex.ac.uk`

1 Sequences, Structures, and Segmenting with Synchrony

Neural Network models have been successful at pattern analysis using just a holistic representation of the input pattern. There is not usually any need to partition the problem or impose strong prior assumptions. This approach has been extended to processing sequences and to processing structures, but with much less success. The storage capacity of a network which is appropriate for short sequences or shallow structures is soon overwhelmed when faced with long sequences or deep structures, leading to much relevant information being lost from the network's representation of state. The solution advocated here is to segment the representation of state into multiple components. The number of components can grow with the length of the sequence or the size of the structure, thereby allowing the capacity required for each component to remain constant. Given this approach, the questions are how to represent these components, on what basis to segment the state information, and what are the implications for learning and processing.

In cognitive neuroscience one proposal for pattern segmentation is that the brain does this segmentation and represents the result using the temporal synchrony of neuron activation spikes [von der Malsburg and Buhmann, 1992]. Any two features of the input which are associated with the same entity are represented using synchronous (or more generally, correlated) activation spikes, and any two features which are associated with different entities are represented with non-synchronous (uncorrelated) activation spikes [von der Malsburg, 1981] [Eckhorn et al., 1988] [Gray et al., 1989] [Shastri and Ajjanagadde, 1993] [Singer and Gray, 1995]. This approach has been applied to visual object recognition [Wang and Terman, 1997] and auditory source recognition [Wang, 1996]. In both domains there are multiple entities in the world which are together producing the sensory input, and the brain needs to segment the sensory input according to which entity is responsible for which aspects of it.

This neuroscientific proposal suggests answers to two of our questions. First, that the synchrony of spiking neurons can be used to represent the components of the segmented state. Second, that this segmentation should be done on the basis of the entities in the domain. The work discussed in the remainder of this abstract looks at our third question, the implications of these two proposals for learning and processing. We argue that these two proposals are intimately related for computational reasons. Indeed, it may be that the entire concept of an entity is an artifact of our brains using temporal synchrony to segment our representation of state.

2 Temporal Synchrony for Artificial Neural Networks

To apply the idea of using the synchrony of neuron activation spikes to a computational architecture, we first need to define an abstraction of the neurobiology which captures only those properties which are relevant to our computational investigation. For this work we use the following abstraction:[1]

---

[1] This abstract model is based on the SHRUTI model of reflexive reasoning [Shastri and Ajjanagadde, 1993] [Shastri, 1999].

Time is divided into discrete periods, and each period is divided into discrete phases.

There is no bound on the number of periods and no bound on the number of phases in a period.

The ordering between phases within a period cannot be used to represent information.

Units are divided into two types, those which process each phase independently (called pulsing units) and those which output the same activation to all phases (called nonpulsing units).

As is commonly done, a unit is intended to correspond to a collection of neurons, so it is possible for a single unit to be pulsing in multiple phases during the same period. Being abstractions, it is possible to question each of these assumptions,[2] but they are appropriate here because they allow us to focus on the primary concern, the use of temporal synchrony.

Simple Synchrony Networks (SSNs) [Lane and Henderson, 1998] [Henderson and Lane, 1998] are an artificial neural network architecture which uses this abstract model of temporal synchrony in a way which supports learning. They are an extension of Simple Recurrent Networks (SRNs) [Elman, 1991], which gives them the ability to process sequences. Each position in the sequence is represented as a period in the network's temporal pattern of activation. The difference between SSNs and SRNs is in the representation of the network's state at each sequence position. SRNs use a holistic representation of state, while SSNs segment this state into information about a set of entities.[3] Each entity is represented by a phase in the network's temporal pattern of activation.

Thus the information about an entity at a sequence position is represented by a pattern of activation in a phase in a period. As illustrated in figure 1, this pattern of activation is computed from holistic inputs at the sequence position, inputs about the entity at the sequence position, and the pattern of activation for the entity during the previous sequence position. The links which perform this computation can be trained using a simple extension of Backpropagation Through Time [Lane and Henderson, 1998].
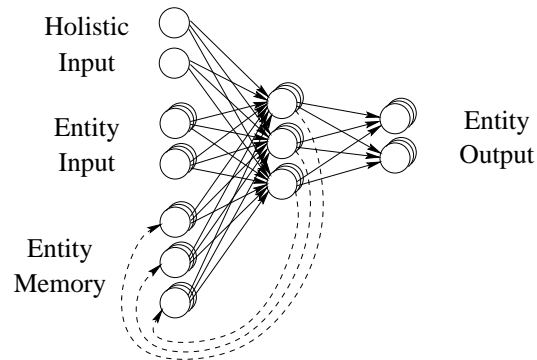

Figure 1: An example of a SSN network. Nonpulsing units are shown as circles, pulsing units as stacks of circles, and links as solid arrows. The dashed arrows represent the copying of activation from the previous period.

3   Generalization with Temporal Synchrony

The motivation for segmenting state information into components was that this would allow us to vary the number of components, and thus the capacity of the state, without having to vary the capacity requirements of each individual component. This requirement is met by the above neural network model because the number of phases can vary without requiring additional training.

---

[2] For example, see [Lane and Henderson, in press] for a discussion of bounding the number of phases in a period.

[3] Some versions of the SSN architecture maintain a holistic representation of state as well as an entity based representation. Without loss of generality, we will talk in terms of the simpler architecture, called type A in [Lane and Henderson, 1998], which has holistic inputs but no holistic state.

These additional phases are simply processed with the same link weights as all the other phases. This processing strategy has significant implications for learning. Anything learned about a computation performed in one phase will automatically be generalized to computations in all other phases. This generalization across phases is an inherent property of using temporal synchrony, since link weights do not change at the time scale of phases.

Processing each phase in the same way and therefore generalizing learned information across phases poses no problems if we use phases to store information about individual entities. Indeed this is exactly what we want. For example, if we learn to recognize that one snake is poisonous then we want to generalize that knowledge to other snakes, even in situations where there are multiple snakes. In language processing, if we learn something about one noun phrase then we want to generalize that knowledge to other noun phrases, even in sentences that have multiple noun phrases. These types of regularities are central to what we mean by "entity", and they are captured by using temporal synchrony to represent entities [Henderson, 1996]. It could even be argued that the concept of entity is derivable from the computational properties of using temporal synchrony, but that argument is beyond the scope of this abstract.

4 An Application to Language Processing

One application which involves both arbitrarily long sequences and arbitrarily large structures is parsing natural language. There is wide variation in the length of sentences and the size of their syntactic structures, and thus wide variation in the capacity required to represent the state of a parser as it processes a sentence. Many attempts have been made to apply holistic neural network representations to parsing, but in one way or another they all run into the problem of a network's limited storage capacity [Ho and Chan, 1999]. As advocated above, this limitation has been overcome using Simple Synchrony Networks [Lane and Henderson, in press]. The SSN's phases are used to store the information about syntactic constituents, thereby segmenting the parser's state into an arbitrarily large set of constituents and allowing each constituent's representation to be of constant size. It also means the network has the linguistically desirable property of generalizing over constituents, as well as simplifying the parser's output representation [Henderson and Lane, 1998].

Recent experiments demonstrate that a SSN parser can handle varying sentence lengths and generalize appropriately, even with the complexities of real natural language. Henderson [2000] compares the performance of a SSN parser to the standard statistical parsing technique of Probabilistic Context Free Grammars. Both models were trained on the same set of sentences from a corpus of naturally occurring text, and then tested on a held out set. The test set contained sentences ranging from 2 words to 59 words in length. The SSN parser output 65% of the target constituents (recall) and 65% of the constituents which it output were in the target (precision). The Probabilistic Context Free Grammar was much less robust, not finding parses for half the sentences, and even in the parsed half only outputting 57% of the target constituents and having 54% of the output constituents be in the target.

5 Conclusions

These empirical results show that segmenting state information on the basis of entities is an effective way to allow neural networks to store arbitrarily large amounts of state information. They also show that using temporal synchrony to represent this segmentation results in appropriate generalization across entities when learning. Thus the neuroscientific proposal of using the synchrony of activation spikes to segment patterns according to entities has been shown to have significant computational implications for artificial neural networks. Simple Synchrony Networks are an artificial neural network architecture which demonstrates the promise of this proposal.

References

[Eckhorn et al., 1988] Eckhorn, R., Bauer, R., Jordan, W., Brosch, M., Kruse, W., Munk, M., and Reitboeck, H. (1988). Coherent oscillations: a mechanism of feature linking in the visual cortex? *Biological Cybernetics*, 60:121–130.

[Elman, 1991] Elman, J. L. (1991). Distributed representations, simple recurrent networks, and grammatical structure. *Machine Learning*, 7:195–225.

[Gray et al., 1989] Gray, C., Koenig, P., Engel, A., and Singer, W. (1989). Oscillatory responses in cat visual cortex exhibit intercolumnar synchronization which reflects global stimulus properties. *Nature*, 338:334–337.

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

[Henderson, 2000] Henderson, J. (2000). A neural network parser that handles sparse data. In *Proceedings of the 6th International Workshop on Parsing Technologies*, pages 123–134, Trento, Italy.

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

[Ho and Chan, 1999] Ho, E. and Chan, L. (1999). How to design a connectionist holistic parser. *Neural Computation*, 11(8):1995–2016.

[Lane and Henderson, 1998] Lane, P. and Henderson, J. (1998). 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, Skovde, Sweden.

[Lane and Henderson, in press] Lane, P. and Henderson, J. (in press). Incremental syntactic parsing of natural language corpora with simple synchrony networks. *IEEE Transactions on Knowledge and Data Engineering*.

[Shastri, 1999] Shastri, L. (1999). Advances in shruti – a neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony. *Applied Intelligence*, 11:79–108.

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

[Singer and Gray, 1995] Singer, W. and Gray, C. (1995). Visual feature integration and the temporal correlation hypothesis. *Annual Review of Neuroscience*, 18:555–586.

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

[von der Malsburg and Buhmann, 1992] von der Malsburg, C. and Buhmann, J. (1992). Sensory segmentation with coupled neural oscillators. *Biological Cybernetics*, 67:233–242.

[Wang, 1996] Wang, D. (1996). Primitive auditory segregation based on oscillatory correlation. *Cognitive Science*, 20:409–456.

[Wang and Terman, 1997] Wang, D. and Terman, D. (1997). Image segmentation based on oscillatory correlation. *Neural Computation*, 9:805–836 (For errata see Neural Computation, 9:1623–1626, 1997).