TEMECOR: An Associative, Episodic, Temporal Sequence Memory*

Gerard J. Rinkus
Department of Cognitive and Neural Systems
Boston University, 111 Cummington Street, Boston, MA 02215
email: rinkus@cns.bu.edu

Abstract

A distributed associative neural model of *episodic memory* for spatio-temporal patterns is presented. The model exhibits *faster-than-linear* capacity scaling, under single-trial learning, for both uncorrelated and correlated patterns. The correlated pattern sets used in simulations reported herein are formally, sets of *complex state sequences* (CSSs)—i.e. sequences in which states can recur multiple times. Efficient representation of large sets of CSSs is central to speech and language processing. The English lexicon, for example, is formally representable as a set of many thousands of CSSs over an alphabet of about 50 phonemes. The model chooses internal representations (IRs) for each state in a highly random fashion. This implies maximal *dispersion*—i.e. maximal average Hamming distance—over the set of IRs chosen during learning. Maximal dispersion yields maximal episodic availability of the traces of the individual exemplars.

1 INTRODUCTION

This paper describes a distributed neural network model, TEMECOR[1], of *episodic memory* for spatio-temporal patterns. Episodic or *autobiographical* memory (Tulving, 1972) is memory for specific events that one has experienced. They are detail-rich memories that can generally last a lifetime even though they are derived from events that occur only once (i.e. single-trial learning). In addition, assuming episodic memories are en-

*This is a substantially revised and extended version of a paper submitted to the World Congress on Neural Networks, 1995, Washington, DC.

[1]Rinkus (1993) contains a Preliminary description of the basic principles of TEMECOR (*Temporal Episodic MEmory using COmbinatorial Representations*).

coded in terms of high-level features, the set of one's episodic memories may contain substantial featural overlap (non-orthogonality).

A special case of the class of spatio-temporal patterns is that which Guyon, Personnaz & Dreyfus (1988) have referred to as *complex state sequences* (CSSs). A CSS is a sequence in which states can recur multiple times—e.g. [A B B C A D B]. The ability to efficiently represent and process CSSs lies at the heart of speech and language modelling. The spoken *lexicon* of English, for example, can be formally represented as a set of on the order of 100,000 CSSs (word forms)[2] over a set (alphabet) of about 50 phonemes (states). Assuming the average number of phonemes per word is five, this means the each phoneme occurs, on average, 10,000 times over the entire lexicon.

We show in section 4 that TEMECOR's capacity increases *faster-than-linearly* in the number of cells in the model. This is achieved with single presentations of each pattern and for both uncorrelated patterns and correlated patterns (CSS case).

In order to successfully represent sets of CSSs like:

Seq. 1: [A B B C A D B]
Seq. 2: [B C B B D A A]
Seq. 3: [A C A B B E]

a model must find different (although possibly overlapped) internal representations (IRs), not only for all instances of a given state, X, *within* each individual sequence, but for all instances of X across *all* sequences in the set; otherwise, during recall, the model will not be able to reliably transition to the correct states following the various instances of state X.

Fig. 1 depicts the basic format of TEMECOR's internal representations of states. The cell groups (a-e) correspond to features. Any cell in the group can be used, in a particular instance, to represent the feature. A state, X, is defined as a set of co-active features. An internal representation of X, IR(X), is a choice of a particular *combination* of cells—one cell in each group corresponding to one of X's features. Thus, fig. 1 depicts one particular IR for the state consisting of features, {a,b,c}. The total number of unique IRs for the state is ($4^3 = 64$). It is the exponentially large IR-space for any particular state that underlies TEMECOR's great capacity for storing CSSs.

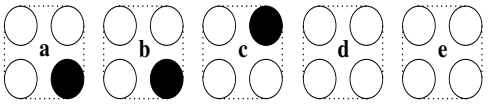

Figure 1: *Depiction of the basic representation format used in TEMECOR. The pattern (i.e. state) has three features, {a,b,c}. Assuming one cell is chosen to represent the corresponding feature in each group, there are* $4^3 = 64$*—i.e. an* exponentially *large number of—unique representations of that state.*

On each timeslice, specific cells, within each active featural group (see fig. 1), are chosen at random. This guarantees that the IRs the model chooses are uniformly distributed in IR-space. While the exact same IR could be chosen for a given state on more than one occasion, this probability is exceedingly small. Specifically,

$$p(IR(X_1) = IR(X_2)) = K^{-S}.$$

[2]This estimate of 100,000 refers to *word forms* (e.g. "jump", "jumped", "jumper", etc. are different forms of the same root word) but since they correspond to distinguishable acoustic patterns (with different meanings), they must all be counted as separate entries.

where X_1 and X_2 denote different instances of state X, S is the number of active features in X, and K is the number of cells per cell group. Furthermore, should such a coincidence occur, it is only the two state sequences in which these two IRs occur that will be affected.

2 RELATED WORK

There have been several recent proposals of recurrent models based on *back-propagation* (Rumelhart, Hinton & Williams, 1986) for processing spatio-temporal patterns—in particular, sets of CSSs generated by a finite-state automaton (FSA) [the SRN (Elman, 1990), the recurrent model of Jordan (1986) which we will call the *Jordan Recurrent Network* (JRN), and the *Real-Time Recurrent Learning* (RTRL) model of Williams & Zipser (1989)]. However, these models have been designed to learn the higher-order statistical regularities (correlational structure) of the set of CSSs. They have not been shown to be capable of episodically recalling the individual exemplars.

The SRN, JRN and RTRL all use continuous-valued cells, and thus have very large IR-spaces. A hidden layer of 15 cells having four resolvable levels of activity has $4^{15} > 1$ billion states (Cleeremans, 1993). The problem, however, is that back-propagation acts to increase the similarity of [or, "homogenize"; Cleeremans (1993)] the chosen IRs. This compression of the *actually-used* regions of IR-space—an effect clearly revealed by hierarchical cluster analysis (Smith & Zipser, 1989; Elman, 1990; Cleeremans, 1993)—underlies these models' powers of generalization and categorization but tends to obliterate exactly the temporal context (state history) information needed for an episodic degree of retention. Furthermore, as Cleeremans points out, this effect is only made worse by further learning.

3 DESCRIPTION OF TEMECOR

TEMECOR has two layers as shown in fig. 2. Layer 1 (L1) contains M binary feature detectors. Layer 2 (L2) contains M winner-take-all *competitive modules* (CMs) which are in one-to-one correspondence with the L1 cells. Each CM has K cells. Whenever a particular L1 cell fires, exactly one of the L2 cells in the corresponding CM is chosen winner. Each L2 cell has an excitatory modifiable {0,1}-valued synapse onto every other L2 cell (except for those in its own CM). It is this set of *horizontal connections* in which the chains encoding the temporal aspect of the inputs are embedded. A simple Hebbian learning rule is used. Every L2 cell active at timeslice t increases its weight onto all L2 cells active at $t+1$ unless the weight has already been increased. Each L2 cell has an unmodifiable synapse onto its corresponding L1 cell. The purpose of these top-down (TD) or *reverse* connections is to allow the appropriate L1 pattern to be reinstated when an L2 pattern reads out during recall.

Although TEMECOR does not require it, we assume for simplicity of exposition that all states have the same number S of active features, where $S < M$. The terms "episode," "spatio-temporal feature pattern" and "state sequence" are generally interchangeable in this paper. A typical episode, Φ^i, consisting of three timeslices can be expressed as:

$$\begin{array}{lll} \Phi^i_1\text{: \{a, b, c\}} & & \text{A: \{a, b, c\}} \\ \Phi^i_2\text{: \{d, e, f\}} & \text{or,} & \text{X: \{d, e, f\}} \\ \Phi^i_3\text{: \{g, h, i\}} & & \text{B: \{g, h, i\}} \end{array}$$

where each Φ^i_j denotes a particular timeslice. Lowercase letters denote features. As shown in the righthand representation, unindexed uppercase letters are sometimes used

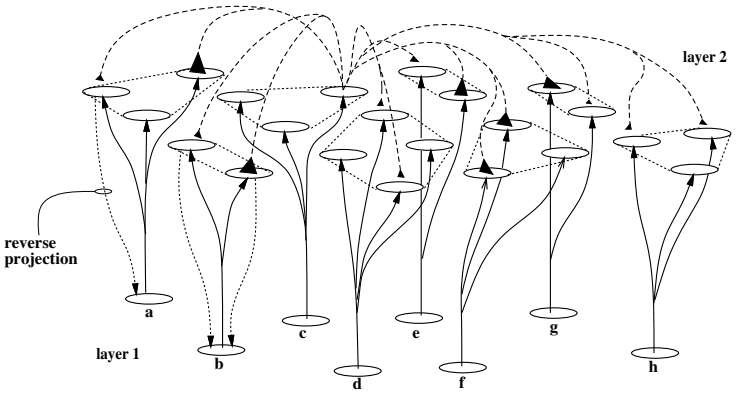

Figure 2: *TEMECOR has two layers. Some of the horizontal connections emanating from one L2 cell are depicted with dashed lines ending in either large (weight = 1) or small (weight = 0) black synapses. Only a few sample reverse (i.e. top-down) projections are shown.*

to represent states.

Fig. 3a shows a particular L2 representation[3] (L2-code) for Φ^i: L2 cells are assumed to be chosen at *random* (in the learning phase) within *active* CMs (i.e. CMs corresponding to active L1 cells), thus ensuring, statistically, a highly dispersed—in terms of Hamming distance—set of L2 codes. L2 codes are denoted with the Greek letter, Δ. The L2 code, Δ^i, corresponding to Φ^i can be written as:

$$\begin{array}{l} \Delta^i_1\text{: } \{a_1, b_2, c_1\} \\ \Delta^i_2\text{: } \{d_2, e_2, f_3\} \\ \Delta^i_3\text{: } \{g_4, h_1, i_3\} \end{array}$$

where the notation, a_1, indicates cell 1 in CM_a.

Fig. 3a also shows the learning that would occur due to presentation of Φ^i. A synapse, w_{xy} is increased to asymptote (i.e. 1) after a single correlation in which cell y is active immediately after cell x. Cell activation levels are {0,1}-valued.

TEMECOR does not require that the set of spatial patterns (timeslices) be orthogonal. Rather, as the challenge we've taken up is to represent sets of CSSs, whole states can recur exactly without presenting a problem to the model. To see this, suppose a second episode, Φ^j, defined as:

$$\begin{array}{lll} \Phi^j_1\text{: \{a, b, k\}} & & \text{C: \{a, b, k\}} \\ \Phi^j_2\text{: \{d, e, f\}} & \text{or,} & \text{X: \{d, e, f\}} \\ \Phi^j_3\text{: \{g, h, n\}} & & \text{D: \{g, h, n\}} \end{array}$$

is presented to the model. Fig. 3b shows one possible L2 trace that could be chosen for Φ^j. Again, L2 winners are chosen at random within active CMs. Both episodes have the same middle state (i.e. $\Phi^i_2 = \Phi^j_2$) as well as a great deal of featural overlap on the other two timeslices. Nevertheless, the L2-code of that middle state is very different in the two instances. In fact, $\left|\Delta^i_2 \cap \Delta^j_2\right| = 1$. This suggests we can prevent interference between the

[3]The L2 codes correspond to the IRs discussed in the introduction.

two memory traces by requiring that a cell have at least Θ active, large (i.e. a weight of 1) synapses in order to fire. The parameter, Θ, is called the *recall threshold*.[4]

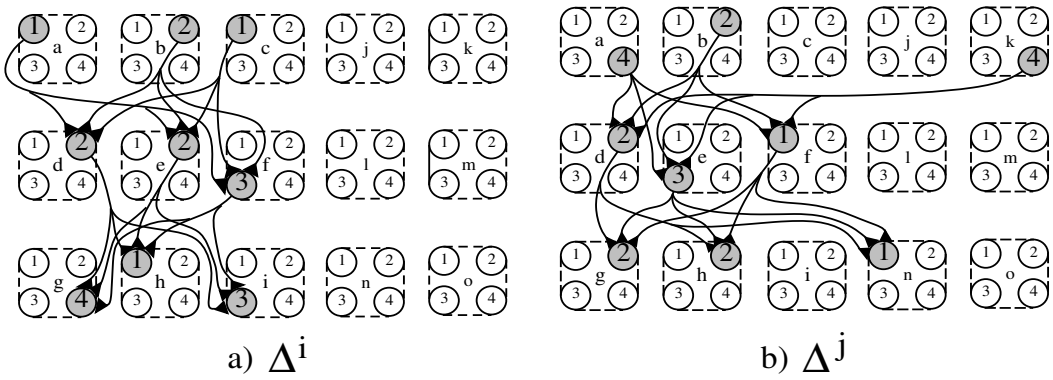

Figure 3: *Active cells are shaded. The three rows correspond to three consecutive time steps. a) a particular L2 code, Δ^i, that might be chosen at random for Φ^i, as well as the corresponding learning. b) an L2 code, Δ^j, for Φ^j.*

Fig. 4 shows that Φ^i is recalled perfectly if $S \geq \Theta \geq 2$. Cells e_3 and f_1 receive input only from b_2 and so do not meet Θ and remain inactive. Cells, d_2, e_2 and f_3, receive three large inputs and correctly become active on the second timeslice of this recall trial. Similarly, none of g_2, h_2 and n_1, become active at $t = 3$ because none of them meets Θ.

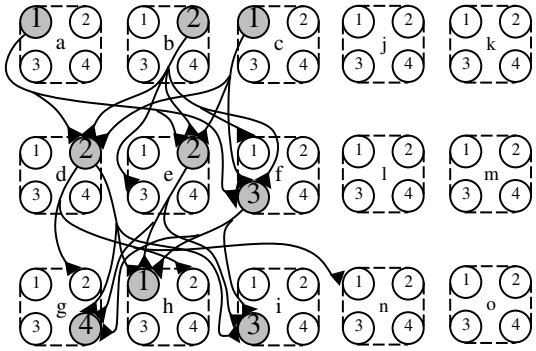

Figure 4: *Recall of Φ^i in the case of $2 \leq \Theta \leq 3$. If $\Theta = 1$, then n_1 would become active at $t = 3$. If $\Theta > 3$, then no recall at all is possible.*

The recall example of fig. 4 assumes a precise *L2 code*—$\Delta^i_1 = \{a_1, b_2, c_1\}$—is provided as a recall prompt. In reality, prompts are necessarily L1 codes; afterall, L1 is the input layer. Using L2-codes rather than L1-codes as recall prompts is justifiable here because it has minor bearing on capacity which is the focus of this paper. However, a more complete version of the model, which provides a mechanism whereby episode-initial L1 patterns cause the correct episode-initial L2 codes to become active, is developed in Rinkus (1995, Ph.D. thesis, in prep.).

[4]The basic idea of this type of *combinatorial memory* is eloquently described, for the spatial pattern domain, in Willshaw, Buneman, & Longuet-Higgins (1969). The models of Marr (1969), Lynch (1986), and Miller (1991) are also instances of spatial combinatorial memories.

Table 1: *Results of simulations using uncorrelated patterns. All Simulations had* $\Theta = 19$, $S = 20$ *and* $T = 10$. *Abbrevs.:* $\hat{F}$ = *ave. instances of each feature, across entire set of episodes;* K = *CM size;* L = *total L2 cells;* $\hat{V}$ = *ave. number of times each L2 cell is used;* $var(V)$ = *variance of* V; W_{inc} = *total number of increased weights;* R_{set} = *recall accuracy over the whole set of episodes; and* H = *percent of horizontal weights increased.*

E	E/L	$\hat{F}$	K	L	$\hat{V}$	$var(V)$	W_{inc}	R_{set}	H
129.3	0.162	258.7	8	800	32.33	42.78	327465.7	97.8	51.7
290.3	0.242	580.7	12	1200	48.39	58.61	736201.3	96.6	51.6
517.0	0.323	1034.0	16	1600	64.62	78.58	1309367.0	97.2	51.7
793.0	0.397	1586.0	20	2000	79.3	72.69	2020240.0	97.2	51.0
1141.7	0.476	2283.3	24	2400	95.14	86.94	2908397.0	97.1	51.0
1544.7	0.552	3089.3	28	2800	110.33	98.93	3942919.0	97.2	50.8
2002.3	0.626	4004.7	32	3200	125.15	115.21	5122921.3	97.1	50.5
2506.0	0.696	5012.0	36	3600	139.22	177.57	6433038.0	97.1	50.1
3084.0	0.771	6168.0	40	4000	154.2	193.86	7925805.0	97.7	50.0

4 SIMULATION RESULTS

Table 4 gives the maximal capacity (and other statistics) for networks of increasing size, in the case of uncorrelated patterns, and where all episodes had $T = 10$ timeslices and each timeslice had $S = 20$ (out of $M = 100$) active features, chosen at random. The product, $T \times S$, will be referred to as the *spatio-temporal complexity* (STC) of an episode.

Parameters constant for all simulations reported in this paper are: $M = 100$, $S = 20$, and the criterion recall accuracy, $R_c = 97.0\%$. Θ, is set to $S-1 = 19$ for all these simulations. Because the degree of overlap over the set of L2 codes increases as additional episodes are presented, maximal capacity is achieved by setting Θ as high as possible (but necessarily less than S).

Table 4 was generated as follows. For each K, the maximal number, E, of episodes which could be stored to criterion accuracy was determined.[5] Recall accuracy, $R(e)$, for a given episode e, is defined as:

$$R(e) = \frac{C(e) - D(e)}{C(e) + I(e)}$$

where $C(e)$ is the number of L2 cells that are correctly active during recall of e, $D(e)$ is the number of *deleted* L2 cells, and $I(e)$ is the number of *intruding* L2 cells. Recall accuracy for a whole set of episodes, R_{set}, is just the average of the R values. All episodes were presented only once.

Table 4 supports the claim that the number of episodes that can be stored to criterion recall accuracy increases faster-than-linearly (at least over the range of network sizes analyzed) in network size. This is also seen in the curves of fig. 5. The second lowest curve (labelled "UNC, STC=200") is derived from table 4. Space limitations prevent inclusion of the tables corresponding to the other seven curves. The curve labelled, "UNC, STC=80", corresponds to simulations in which $T = 4$ and $S = 20$—i.e. a total of 80 featural instances per episode; the simulations giving rise to the curve, "UNC, STC=120", had $T = 6$ and $S = 20$; etc.

The solid curves of fig. 5 show that a slightly slower, although still faster-than-linear,

[5]Each line (i.e. data point) of all tables represents the average of three simulations with the corresponding parameter set.

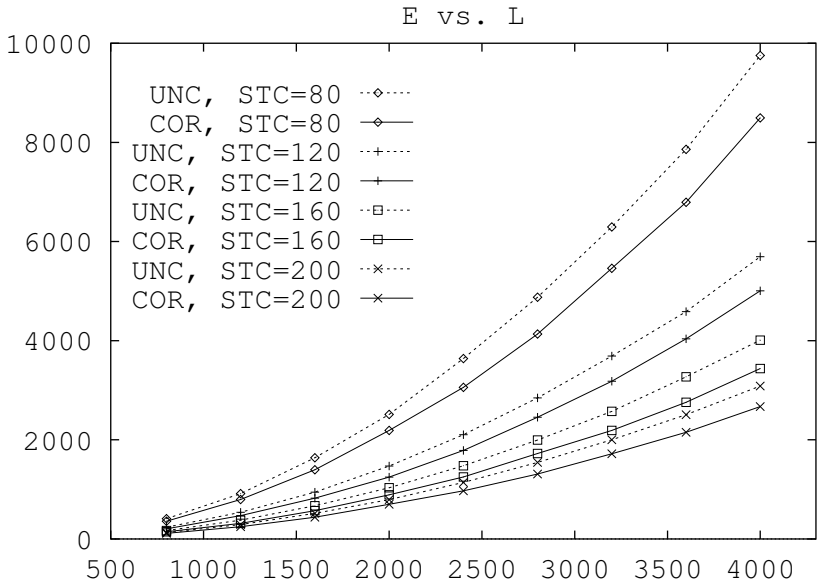

Figure 5: *The four dotted curves correspond to simulations in which uncorrelated episodes of varying STCs—80,120,160 and 200—were used. The solid curves correspond to simulations involving correlated episodes.*

relationship also holds for the case of correlated patterns. The episodes used in the CSS simulations were constructed as follows. First, a set (alphabet) of $U = 100$ unique states, each consisting of 20 active features (chosen randomly from the 100 possible features), was built. The timeslices comprising the episodes were then randomly chosen (with replacement) from this alphabet of 100 states. In the "L=4000" data point of the "COR, STC=80" curve, for example, the average number of occurrences of each state is about 340.

5 CONCLUSION

The simulation results, for both uncorrelated and correlated patterns, show that TEMECOR's capacity scales faster-than-linearly with problem size, while requiring only single-trial learning. This finding is especially encouraging in the CSS case since linguistic information (e.g. the lexicon) can be represented as sets of CSSs over a finite alphabet. In addition, since the horizontal weights do not decrease, *stability* depends only on the degree of weight saturation, not on the frequency or order of occurrence of patterns. This is a necessary property of an episodic memory which is, almost by definition, a repository for statistically rare events. In contrast, models based on back-propagation have been shown to be subject to massive ("catastrophic") forgetting (McCloskey & Cohen, 1989) in which newly encountered patterns obliterate old and infrequently accessed memory traces.

Space does not permit a more detailed discussion of TEMECOR's other properties and

capabilities. However, it retains its faster-than-linear capacity scaling across wide regions of parameter space (e.g. various degrees of horizontal connectivity, varying S, etc.). Furthermore, generalization and categorization properties can be added by generalizing the model so that the choice of the current IR (i.e. L2 code) depends partly on the previous L2 code, partly on the current input (i.e. L1 code), and partly on noise. Such a generalized version of the model, in which the relative mixture of these three influences varies during processing as a function of the difference between expected and actual input, is developed in Rinkus (1995).

References

1. Cleeremans, A. (1993) *Mechanisms of Implicit Learning: Connectionist Models of Sequence Processing*, The MIT Press, Cambridge, MA
2. Elman, J. L. (1990) Finding structure in time. *Cognitive Science*, *14*(2), 179–212.
3. Guyon, I., Personnaz, L., & Dreyfus, G. (1988) Of points and loops. In Eckmiller, R. & Malsburg, C. v.d. (Eds.) *Neural Computers*, NATO ASI Series, Vol. F41, 261–269. Springer-Verlag, Berlin, Germany.
4. Jordan, M. I. (1986) Serial order. Tech. rep. 8604, Institute for Cognitive Science, University of California, San Diego.
5. McCloskey, M. & Cohen, N. J. (1989) Catastrophic interference in connectionist networks: The sequential learning problem. In Bower, G. H. (Ed.), *The Psychology of Learning and Motivation*, v.24, 109–165. Academic Press.
6. Rinkus, G. (1993) Context-sensitive spatio-temporal memory. Proceedings of World Congress on Neural Networks. The annual meeting of the International Neural Network Society. Lawrence Earlbaum Associates, Inc., V.2. 344–347
7. Rinkus, G. J. (1995) *A Combinatorial Neural Netork Exhibiting both Episodic Memory and Generalization for Spatio-Temporal Patterns.* Ph.D. thesis, Graduate School of Arts and Sciences, Boston University. In progress.
8. Rumelhart, D., Hinton, G., & Williams, R. (1986) Learning internal representations by error propagation. in McClelland, J., Rumelhart, D. & the PDP Research Group (Eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Vol. 1, chap. 8, pp. 318-364. The MIT Press.
9. Smith, A. W. & Zipser, D. (1989) Learning sequential structure with the real-time recurrent learning algorithm. *International J. of Neural Systems*, *1*(2), 125–131.
10. Tulving, E. (1972) Episodic and semantic memory. In Tulving, E., & Donaldson, W. (Eds.), *Organization of Memory.* Academic Press, New York.
11. Williams, R. J. & Zipser, D. (1989) A learning algorithm for continually running fully recurrent neural networks. *Neural Computation*, *1*, 270–280.