On Non-Markovian Topographic Organization of Receptive Fields in Recursive Self-Organizing Map

Peter Tiňo[1] Igor Farkaš[2]

[1] School Of Computer Science, University Of Birmingham
Birmingham B15 2TT, UK
[2] Faculty of Mathematics, Physics and Informatics, Comenius University
Mlynská dolina, 842 48 Bratislava, Slovak Republic

Abstract. Recently, there has been an outburst of interest in extending topographic maps of vectorial data to more general data structures, such as sequences or trees. The representational capabilities and internal representations of the models are not well understood. We concentrate on a generalization of the Self-Organizing Map (SOM) for processing sequential data – the Recursive SOM (RecSOM [1]). We argue that contractive fixed-input dynamics of RecSOM is likely to lead to Markovian organizations of receptive fields on the map. We show that Markovian topographic maps of sequential data can be produced using a simple fixed (non-adaptable) dynamic module externally feeding a standard topographic model designed to process static vectorial data of fixed dimensionality (e.g. SOM). We elaborate upon the importance of non-Markovian organizations in topographic maps of sequential data.

1 Introduction

In its original form the self-organizing map (SOM) [2] is a nonlinear projection method that maps a high-dimensional metric vector space onto a two-dimensional regular grid in a topologically ordered fashion. Many modifications of the standard SOM have been proposed in the literature (e.g. [3]). Formation of topographic maps via self-organization constitutes an important paradigm in machine learning with many successful applications e.g. in data and web-mining. Most approaches to topographic map formation operate on the assumption that the data points are members of a finite-dimensional vector space of a fixed dimension. Recently, there has been an outburst of interest in extending topographic maps to more general data structures, such as sequences or trees.

Several modifications of SOM to sequences and/or tree structures have been proposed in the literature ([4] and [5] review most of the approaches). Modified versions of SOM that have enjoyed a great deal of interest equip SOM with *additional feed-back connections* that allow for natural processing of recursive data types. Typical examples of such models are Temporal Kohonen Map [6], recurrent SOM [7], feedback SOM [8], recursive SOM [1], merge SOM [9] and SOM

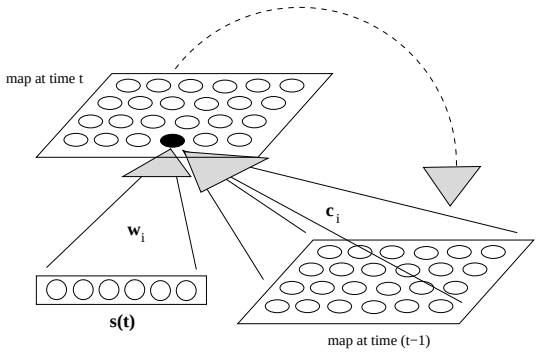

Fig. 1. Recursive SOM architecture.The original SOM algorithm is used for both input vector $\mathbf{s}$(t) and for the context represented as the map activation $\mathbf{y}$(t-1) from the previous time step. Solid lines represent trainable connections, dashed line represents one-to-one copy of the activity vector $\mathbf{y}$. The network learns to associate the current input with previous activity states. This way each neuron responds to a sequence of inputs.

for structured data [10]. However, at present there is still no general consensus as to how best to process sequences with SOMs and this topic remains a very active focus of current neurocomputational research [4, 11, 12].

In this paper, we view such models as non-autonomous dynamical systems with internal dynamics driven by a stream of external inputs. In the line of our recent research, we study the organization of the non-autonomous dynamics on the basis of dynamics of individual fixed-input maps [13]. We concentrate on the Recursive SOM (RecSOM) [1], because RecSOM transcends the simple local recurrence of leaky integrators of earlier models and it has been demonstrated that it can represent much richer dynamical behavior [12]. The principal question driving this research can be stated as: 'What can be gained by having a trainable recurrent part in RecSOM, i.e. how does RecSOM compare with a much simpler setting of SOM operating on a simple *non-trainable* iterative function system with Markovian state-space organization [14]?"

2 Recursive Self-Organizing Map (RecSOM)

The architecture of the RecSOM model [1] is shown in figure 1. Each neuron $i \in \{1, 2, ..., N\}$ in the map has two weight vectors associated with it:

- $\mathbf{w}_i \in \mathbb{R}^n$ – linked with an n-dimensional input $\mathbf{s}(t)$ feeding the network at time t
- $\mathbf{c}_i \in \mathbb{R}^N$ – linked with the context

$$\mathbf{y}(t-1) = (y_1(t-1), y_2(t-1), ..., y_N(t-1))$$

containing map activations $y_i(t-1)$ from the previous time step.

The output of a unit i at time t is computed as

$$y_i(t) = \exp(-d_i(t)), \tag{1}$$

where[1]

$$d_i(t) = \alpha \cdot \|\mathbf{s}(t) - \mathbf{w}_i\|^2 + \beta \cdot \|\mathbf{y}(t-1) - \mathbf{c}_i\|^2. \tag{2}$$

In eq. (2), $\alpha > 0$ and $\beta > 0$ are model parameters that respectively influence the effect of the input and the context upon neuron's profile. Both weight vectors can be updated using the same form of learning rule [1]:

$$\Delta\mathbf{w}_i = \gamma \cdot h_{ik} \cdot (\mathbf{s}(t) - \mathbf{w}_i), \tag{3}$$

$$\Delta\mathbf{c}_i = \gamma \cdot h_{ik} \cdot (\mathbf{y}(t-1) - \mathbf{c}_i), \tag{4}$$

where k is an index of the best matching unit at time t, $k = \mathrm{argmin}_{i\in\{1,2,...,N\}}\, d_i(t)$, and $0 < \gamma < 1$ is the learning rate. Note that the best matching ('winner') unit can be equivalently defined as the unit k of the highest activation $y_k(t)$:

$$k = \underset{i\in\{1,2,...,N\}}{\mathrm{argmax}}\; y_i(t). \tag{5}$$

Neighborhood function h_{ik} is a Gaussian (of width σ) on the distance $d(i,k)$ of units i and k in the map:

$$h_{ik} = e^{-\frac{d(i,k)^2}{\sigma^2}}. \tag{6}$$

The 'neighborhood width' σ decreases in time to allow for forming topographic representation of input sequences.

Under a fixed input vector $\mathbf{s} \in \mathbb{R}^n$, the time evolution (2) becomes

$$d_i(t+1) = \alpha \cdot \|\mathbf{s} - \mathbf{w}_i\|^2 + \beta \cdot \| \left(e^{-d_1(t)}, e^{-d_2(t)}, ..., e^{-d_N(t)}\right) - \mathbf{c}_i\|^2. \tag{7}$$

After applying a one-to-one coordinate transformation $y_i = e^{-d_i}$, eq. (7) reads

$$y_i(t+1) = e^{-\alpha\|\mathbf{s}-\mathbf{w}_i\|^2} \cdot e^{-\beta\|\mathbf{y}(t)-\mathbf{c}_i\|^2}, \tag{8}$$

or, in the vector form:

$$\mathbf{y}(t+1) = \mathbf{F}_{\mathbf{S}}(\mathbf{y}(t)). \tag{9}$$

3 IFS sequence representations combined with standard SOM (IFS+SOM)

Previously, we have shown that a simple affine contractive iterative function system (IFS) [15] can be used to transform temporal structure of symbolic sequences into a spatial structure of points in a metric space [14]. The points represent subsequences in a Markovian manner: Subsequences sharing a common suffix are

[1] $\|\cdot\|$ denotes the Euclidean norm

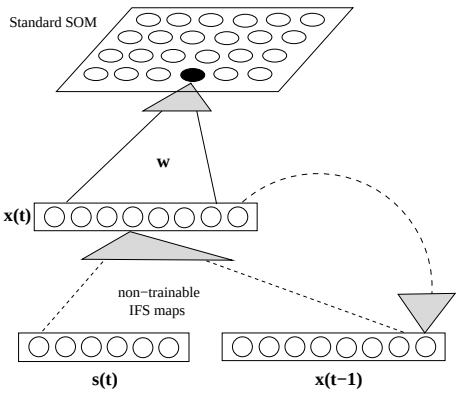

Fig. 2. Standard SOM operating on IFS representations of symbolic streams (IFS+SOM model). Solid lines represent trainable feed-forward connections. No learning takes place in the dynamic IFS part responsible for processing temporal contexts in the input stream.

mapped close to each other. Furthermore, the longer is the shared suffix the closer lie the subsequence representations.

The IFS representing sequences over an alphabet $\mathcal{A}$ of A symbols operates on an m-dimensional unit hypercube $[0,1]^m$, where[2] $m = \lceil \log_2 A \rceil$. With each symbol $s \in \mathcal{A}$ we associate an affine contraction on $[0,1]^m$,

$$s(\mathbf{x}) = k\mathbf{x} + (1-k)\mathbf{t}_s, \quad \mathbf{t}_s \in \{0,1\}^m, \quad \mathbf{t}_s \neq \mathbf{t}_{s'} \text{ for } s \neq s', \tag{10}$$

with contraction coefficient $k \in (0, \frac{1}{2}]$. For a prefix $u = u_1 u_2 ... u_n$ of a string v over $\mathcal{A}$ and a point $\mathbf{x} \in [0,1]^m$, the point

$$u(\mathbf{x}) = u_n(u_{n-1}(...(u_2(u_1(\mathbf{x})))...)) = (u_n \circ u_{n-1} \circ ... \circ u_2 \circ u_1)(\mathbf{x}) \tag{11}$$

constitutes a spatial representation of the prefix u under the IFS (10). Finally, the overall temporal structure of symbols in a (possibly long) sequence v over $\mathcal{A}$ is represented by a collection of the spatial representations $u(\mathbf{x})$ of all its prefixes u, with a convention that $\mathbf{x} = \{\frac{1}{2}\}^m$.

The IFS-based Markovian coding scheme can be used to construct generative probabilistic models on sequences analogous to the variable memory length Markov models [14]. Key element of the construction is a quantization of the spatial IFS representations into clusters that group together subsequences sharing potentially long suffixes (densely populated regions of the suffix-organized IFS subsequence representations).

The Markovian layout of the IFS representations of symbolic sequences can also be used for constructing suffix-based topographic maps of symbolic streams in an unsupervised manner. By applying a standard SOM [16] to the IFS representations one may readily obtain topographic maps of Markovian flavour,

[2] for $x \in \mathbb{R}$, $\lceil x \rceil$ is the smallest integer y, such that $y \geq x$

similar to those obtained by RecSOM. The key difference between RecSOM and IFS+SOM (standard SOM operating on IFS representations) is that the latter approach assumes a fixed non-trainable dynamic part responsible for processing temporal contexts in the input stream. The recursion is not a part of the map itself, but is performed outside the map as a preprocessing step before feeding the standard SOM (see figure 2).

3.1 Relation between IFS+SOM and recurrent SOM

There is a connection between the IFS+SOM and recurrent SOM (RSOM) [7] models. Given a sequence $s_1 s_2 ...$ over a finite alphabet $\mathcal{A}$, the RSOM model determines the winner neuron at time t by identifying the neuron i with the minimal norm of

$$\mathbf{d}_i(t) = \nu \ (\mathbf{t}_{s_t} - \mathbf{w}_i) + (1 - \nu) \ \mathbf{d}_i(t-1), \tag{12}$$

where $0 < \nu < 1$ is a parameter determining the rate of 'forgetting the past', $\mathbf{t}_{s_t}$ is the code of symbol s_t presented at RSOM input at time t and $\mathbf{w}_i$ is the weight vector on connections connecting the inputs with neuron i.

Inputs $\mathbf{x}(t)$ feeding standard SOM in the IFS+SOM model evolve with the IFS dynamics (see (10) and (11))

$$\mathbf{x}(t) = k \ \mathbf{x}(t-1) + (1 - k) \ \mathbf{t}_{s_t}, \tag{13}$$

where $0 < k < 1$ is the IFS contraction coefficient. Best matching unit in SOM is determined by finding the neuron i with the minimal norm of

$$\mathbf{D}_i(t) = \mathbf{x}(t) - \mathbf{w}_i = k \ \mathbf{x}(t-1) + (1 - k) \ \mathbf{t}_{s_t} - \mathbf{w}_i. \tag{14}$$

But $\mathbf{D}_i(t-1) = \mathbf{x}(t-1) - \mathbf{w}_i$, and so

$$\mathbf{D}_i(t) = k \ \mathbf{D}_i(t-1) + (1 - k) \ (\mathbf{t}_{s_t} - \mathbf{w}_i), \tag{15}$$

which, after setting $\nu = 1 - k$, leads to

$$\mathbf{D}_i(t) = \nu \ (\mathbf{t}_{s_t} - \mathbf{w}_i) + (1 - \nu) \ \mathbf{D}_i(t-1). \tag{16}$$

Provided $\nu = 1 - k$, the equations (12) and (16) are equivalent.

The key difference between RSOM and IFS+SOM models lies in the training process. While in RSOM, the best matching unit i with minimal norm of $\mathbf{d}_i(t)$ is shifted towards the current input $\mathbf{t}_{s_t}$, in IFS+SOM the winner unit i with minimal norm of $\mathbf{D}_i(t)$ is shifted towards the (Markovian) IFS code $\mathbf{x}(t)$ coding the whole history of recently seen inputs.

4 Experiments

We compare RecSOM with standard SOM operating on Markovian suffix-based vector representations of fixed dimensionality (IFS+SOM) on a corpus of written

English, the novel "Brave New World" by Aldous Huxley. This data set was used in [1].

In the corpus we removed punctuation symbols, upper-case letters were switched to lower-case and the space between words was transformed into a symbol '-'. The complete data set (after filtering) comprised 356606 symbols. Letters of the Roman alphabet were binary-encoded using 5 bits and presented to the network one at a time. RecSOM with 400 neurons was trained for two epochs using the following parameter settings: $\alpha = 3$, $\beta = 0.7$, $\gamma = 0.1$ and $\sigma : 10 \rightarrow 0.5$. Radius σ reached its final value at the end of the first epoch and then remained constant to allow for fine-tuning of the weights. In the IFS+SOM model, the IFS coefficient was set to $k = 0.3$. Other parameters, such as size of the map, learning rate, and time schedule for reducing the neighborhood width σ were the same as in RecSOM.

We constructed a map of the neurons' receptive fields (RFs) (shown in figure 3). Following [1], RF of a neuron is defined as the common suffix of all sequences for which that neuron becomes the best-matching unit. It is evident that the RFs are topographically ordered with respect to the most recent symbols.

For these RFs, we computed the quantizer depth (according to [1]), which quantifies the amount of memory captured by the map. It is defined as

$$\bar{n} = \sum_{i=1}^{N} p_i n_i, \tag{17}$$

where p_i is the probability of the RF of neuron i, and n_i is its length. The quantizer depth was $\bar{n} = 1.91$.

The RecSOM model can be considered a nonautonomous dynamical system driven by the external input stream (in this case, sequences over the Roman alphabet $\mathcal{A}$). In order to investigate the fixed-input dynamics (9) of the mappings[3] $\mathbf{F}_s$, we randomly (with uniform distribution) initialized context activations $\mathbf{y}(0)$ in 10,000 different positions within the state space $(0, 1]^N$. For each initial condition $\mathbf{y}(0)$, we checked asymptotic dynamics of the fixed input maps $\mathbf{F}_s$ by monitoring L_2-norm of the activation differences $(\mathbf{y}(t) - \mathbf{y}(t-1))$ and recording the limit set (after 1000 iterations). We observed that all autonomous dynamics settle down in the respective unique attractive fixed points $\mathbf{y}_s = \mathbf{F}_s(\mathbf{y}_s)$, $s \in \mathcal{A}$,

It is important to appreciate how the character of the RecSOM fixed-input dynamics (9) for each individual input symbol $s \in \mathcal{A}$ shapes the overall organization of the map. For each input symbol s, the autonomous dynamics $\mathbf{y}(t) = \mathbf{F}_s(\mathbf{y}(t-1))$ induces a dynamics of the winner units on the map:

$$i_s(t) = \underset{i \in \{1,2,\ldots,N\}}{\operatorname{argmax}} \; y_i(t). \tag{18}$$

The dynamics (18) is illustrated in figure 4 (left). For each of the 10,000 initial conditions $\mathbf{y}(0)$, we first let the system (9) settle down by preiterating it

[3] We slightly abuse the mathematical notation here. As arguments of the bounds, we write the actual input symbols, rather than their vector encodings $\mathbf{s}$.

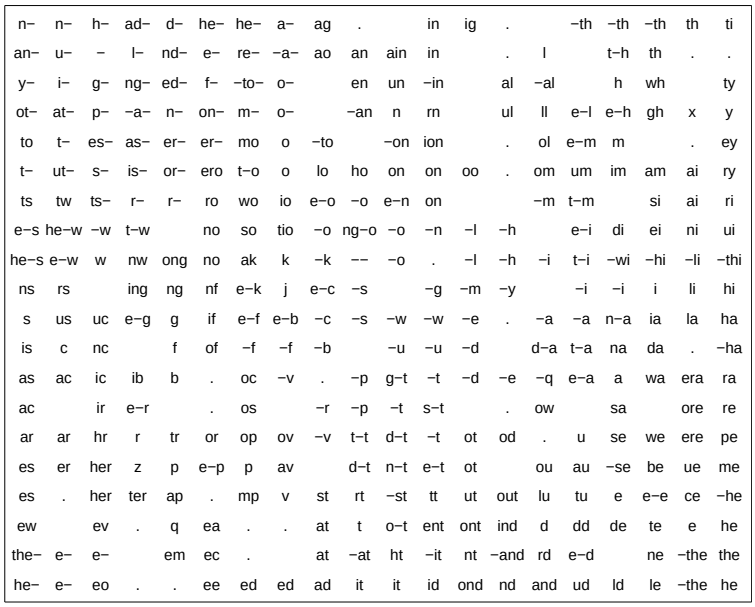

Fig. 3. Receptive fields of RecSOM trained on English text. Dots denote units with empty RFs.

for 1000 iterations and then mark the map position of the winner units $i_s(t)$ for further 100 iterations. If the fixed-input dynamics for $s \in \mathcal{A}$ is dominated by the unique attractive fixed point $\mathbf{y}_s$, the induced dynamics on the map, (18), settles down in neuron i_s, corresponding to the mode of $\mathbf{y}_s$:

$$i_s = \underset{i \in \{1,2,\ldots,N\}}{\operatorname{argmax}} \; y_{s,i}. \tag{19}$$

The neuron i_s will be most responsive to input subsequences ending with long blocks of symbols s. Such an organization follows from the attractive fixed point behaviour of the individual maps $\mathbf{F}_s$, $s \in \mathcal{A}$, and the unimodal character of their fixed points $\mathbf{y}_s$. As soon as symbol s is seen, the mode of the activation profile $\mathbf{y}$ drifts towards the neuron i_s. The more consecutive symbols s we see, the more dominant the attractive fixed point of $\mathbf{F}_s$ becomes and the closer the winner position is to i_s. This mechanism for creating suffix-based RF organization is reminiscent of the Markovian fractal subsequence representations used in our IFS+SOM model.

We observed a variety of asymptotic regimes of the fixed-input RecSOM dynamics (9). For some symbols, the fixed-input dynamics converges to an attractive fixed point; for other symbols (e.g. symbols 'i', 't', 'a', '-'), the dynamics followed a period-two attractor. Fixed input RecSOM dynamics for symbols 'e' and 'o' followed a complicated a-periodic trajectory.

Dynamics of the winner units on the map induced by the fixed-input dynamics of $\mathbf{F}_s$ are shown in figure 4 (left). For symbols s with dynamics $\mathbf{y}(t) =$

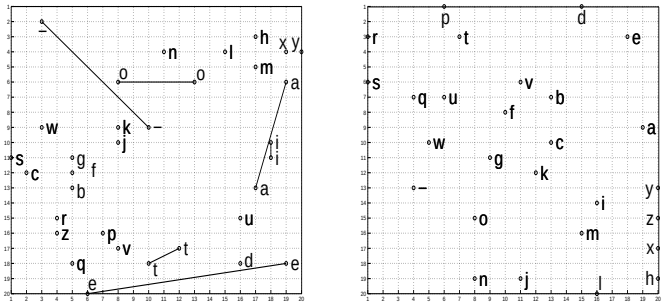

Fig. 4. Dynamics of the winning units on the RecSOM (left) and IFS+SOM (right) maps induced by the fixed-input dynamics. The maps were trained on a corpus of written English ("Brave New World" by Aldous Huxley).

$\mathbf{F}_s(\mathbf{y}(t-1))$ dominated by a single fixed point $\mathbf{y}_s$, the induced dynamics on the map settles down in the mode position of $\mathbf{y}_s$. However, some autonomous dynamics $\mathbf{y}(t) = \mathbf{F}_s(\mathbf{y}(t-1))$ of period two (e.g. $s \in \{n, h, r, p, s\}$) induce a trivial dynamics on the map driven to a single point (grid position). In those cases, the points $\mathbf{y}^1$, $\mathbf{y}^2$ on the periodic orbit ($\mathbf{y}^1 = \mathbf{F}_s(\mathbf{y}^2)$, $\mathbf{y}^2 = \mathbf{F}_s(\mathbf{y}^1)$) lie within the representation region (Voronoi compartment) of the same neuron. Interestingly enough, the complicated dynamics of $\mathbf{F}_o$ and $\mathbf{F}_e$ translates into aperiodic oscillations between just two grid positions. Still, the suffix based organization of RFs in figure 3 is shaped by the underlying collection of the fixed input dynamics of $\mathbf{F}_s$ (illustrated in figure 4 (left) through the induced dynamics on the map).

The IFS+SOM map ($k = 0.3$) is shown in figure 5 (quantizer depth $\bar{n} = 1.69$). The induced dynamics on the map is illustrated in figure 4 (right). The suffix based organization of RFs is shaped by the underlying collection of autonomous attractive IFS dynamics.

5 Discussion

Periodic (beyond period 1), or aperiodic attractive dynamics of autonomous systems $\mathbf{y}(t) = \mathbf{F}_s(\mathbf{y}(t-1))$ lead to potentially complicated non-Markovian organizations of RFs on the map. By calculating the RF of a neuron i as the common suffix shared by subsequences yielding i as the best matching unit [1], we always create a suffix based map of RFs. Such RF maps are designed to illustrate the temporal structure learned by RecSOM. Periodic or aperiodic dynamics of $\mathbf{F}_s$ can result in a 'broken topography' of RFs: two sequences with the same suffix can be mapped into distinct positions on the map, separated by a region of very different suffix structure. For example, depending on the context, subsequences ending with 'ee' can be mapped either near the lower-left, or near the lower-right corners of the RF map in figure 3. Unlike in contractive RecSOM or IFS+SOM models, such context-dependent RecSOM maps embody a potentially unbounded memory structure, because the current position of the

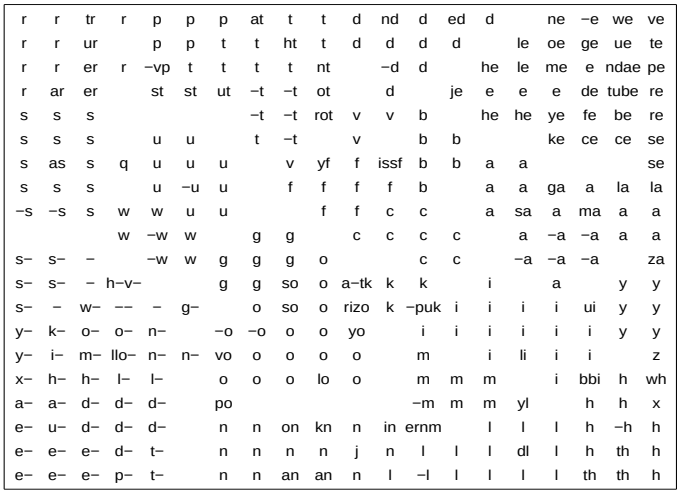

Fig. 5. Receptive fields of a standard SOM with 20×20 units trained on IFS outputs, obtained on the English text. Topographic organization is observed with respect to the most recent symbols.

winner neuron is determined by the whole series of processed inputs, and not only by a history of recently seen symbols. Unless we understand the driving mechanism behind such context-sensitive suffix representations, we cannot fully appreciate the meaning of the RF structure of a RecSOM map.

One has to ask what is the principal motivation behind building topographic maps of sequential data? If the motivation is a better understanding of cortical signal representations (e.g. [17]), then a considerable effort should be devoted to mathematical analysis of the scope of potential temporal representations and conditions for their emergence. If, on the other hand, the primary motivation is data exploration or data preprocessing, then we need to strive for a solid understanding of the way temporal contexts get represented on the map and in what way such representations fit the bill of the task we aim to solve.

There will be situations, where finite memory Markovian context representations are quite suitable. In that case, contractive RecSOM models, and indeed IFS+SOM models as well, may be appropriate candidates. But then the question arises of why exactly there needs to be a *trainable* dynamic part in self-organizing maps generalized to handle sequential data. For more complicated data sets, like the English language corpus, RF maps beyond simple Markovian organization may be preferable. Yet, it is crucial to understand exactly what structures that are more powerful than Markovian organization of RFs are desired and why. It is appealing to notice in the RF map of figure 3 the clearly non-Markovian spatial arrangement into distinct regions of RFs ending with the word-separation symbol '-'. Because of the special role of '-' and its high frequency of occurrence, it may indeed be desirable to separate endings of words in distinct islands with

more refined structure. However, to go beyond mere commenting on empirical observations, one needs to address issues such as

- what properties of the input stream are likely to induce periodic (or aperiodic) fixed input dynamics leading to context-dependent RF representations in SOMs with feedback structures,
- what periods for which symbols are preferable,
- what is the learning mechanism (e.g. sequence of bifurcations of the fixed input dynamics) of creating more complicated context dependent RF maps.

Those are the challenges for our future work.

References

1. Voegtlin, T.: Recursive self-organizing maps. Neural Networks **15** (2002) 979–992
2. Kohonen, T.: Self–organizing formation of topologically correct feature maps. Biological Cybernetics **43** (1982) 59–69
3. Yin, H.: ViSOM - a novel method for multivariate data projection and structure visualisation. IEEE Transactions on Neural Networks **13** (2002) 237–243
4. de A. Barreto, G., Araújo, A., Kremer, S.: A taxanomy of spatiotemporal connectionist networks revisited: The unsupervised case. Neural Computation **15** (2003) 1255–1320
5. Hammer, B., Micheli, A., Strickert, M., Sperduti, A.: A general framework for unsupervised processing of structured data. Neurocomputing **57** (2004) 3–35
6. Chappell, G., Taylor, J.: The temporal kohonen map. Neural Networks **6** (1993) 441–445
7. Koskela, T., znd J. Heikkonen, M.V., Kaski, K.: Recurrent SOM with local linear models in time series prediction. In: 6th European Symposium on Artificial Neural Networks. (1998) 167–172
8. Horio, K., Yamakawa, T.: Feedback self-organizing map and its application to spatio-temporal pattern classification. International Journal of Computational Intelligence and Applications **1** (2001) 1–18
9. Strickert, M., Hammer, B.: Neural gas for sequences. In: Proceedings of the Workshop on Self-Organizing Maps (WSOM'03). (2003) 53–57
10. Hagenbuchner, M., Sperduti, A., Tsoi, A.: Self-organizing map for adaptive processing of structured data. IEEE Transactions on Neural Networks **14** (2003) 491–505
11. Schulz, R., Reggia, J.: Temporally asymmetric learning supports sequence processing in multi-winner self-organizing maps. Neural Computation **16** (2004) 535–561
12. Hammer, B., Micheli, A., Sperduti, A., Strickert, M.: Recursive self-organizing network models. Neural Networks **17** (2004) 1061–1085
13. Tiňo, P., Čerňanský, M., Beňušková, L.: Markovian architectural bias of recurrent neural networks. IEEE Transactions on Neural Networks **15** (2004) 6–15
14. Tiňo, P., Dorffner, G.: Predicting the future of discrete sequences from fractal representations of the past. Machine Learning **45** (2001) 187–218
15. Barnsley, M.: Fractals everywhere. Academic Press, New York (1988)
16. Kohonen, T.: The self-organizing map. Proceedings of the IEEE **78** (1990) 1464–1479
17. Wiemer, J.: The time-organized map algorithm: Extending the self-organizing map to spatiotemporal signals. Neural Computation **16** (2003) 1143–1171