領域表現を用いたKFM連想メモリにおけるアナログ時系列パターンの学習

白鳥友規　長名優子

東京工科大学 コンピュータサイエンス学部

1 はじめに

近年，生物の脳や神経系に見られるような柔軟な情報処理を行う手法として，ニューラルネットワークの研究が盛んに行われており，多くのモデルが提案されている．しかしながら，これらのモデルの多くでは学習過程と実行過程が分離しているため，学習すべき情報があらかじめすべて与えられていなければ学習を行うことができない．それに対し，実際にはあらかじめ記憶すべき情報がすべては得られない場合も数多く存在する．そのような場合には学習過程と実行過程を区別しない逐次学習可能なモデルが必要となる．文献 [4] では，逐次学習が可能な時系列パターンの連想を扱うことが可能なモデルとして時系列パターンのための領域表現を用いた KFM(Kohonen Feature Map) 連想メモリが提案されているが，このモデルではアナログパターンを扱うことができない．

本研究では，時系列パターンのための領域表現を用いた KFM 連想メモリをアナログパターンの連想が扱えるように改良したモデルを提案する．

2 アナログ時系列パターンのための領域表現を用いた KFM 連想メモリ

2.1 構造

提案モデルは，図 1 に示すような入出力層とマップ層の 2 層から構成されている．

2.2 学習過程

$\boldsymbol{Y}^{(k,1)} \to \boldsymbol{Y}^{(k,2)} \to \cdots \to \boldsymbol{Y}^{(k,t_k)}$ を学習させる場合を考える．時系列パターンのための KFM 連想メモリでは，k 番目の時系列パターンの t 番目のパターンを学習するための学習ベクトル $\{\boldsymbol{X}^{(k,t)}\}_{k=1,\cdots,p}$ として以下のようなベクトル用いる．

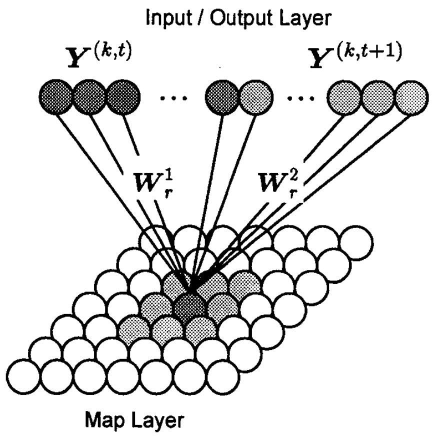


図 1: アナログ時系列パターンのための領域表現を用いた KFM 連想メモリの構造

$$\boldsymbol{X}^{(k,t)} = \begin{pmatrix} \boldsymbol{Y}^{(k,t)} \\ \boldsymbol{0} \end{pmatrix} + \begin{pmatrix} \boldsymbol{0} \\ \boldsymbol{Y}^{(k,t+1)} \end{pmatrix} \qquad (1)$$

時系列パターンのための KFM 連想メモリでは，重みを以下の手順に従って更新することにより学習を行う．

(1) 重みの初期値をランダムに選び，再帰差分ベクトルを $\boldsymbol{y_i} = \boldsymbol{0}$ とする．

(2) マップ層の各ニューロンに対して，再帰差分ベクトルを求める．n 回目の学習の繰り返しにおける時系列パターン中の t 番目のパターンに対するマップ層ニューロン i の再帰差分ベクトル $\boldsymbol{y_i}(n,t)$ は

$$y_{ij}(n,t) = \begin{cases} (1-\beta)y_{ij}(n,t-1) \\ \quad +\beta(X_j{}^{(k,t)} - W_{ij}(n,t)) \ (1 \le j \le N/2) \\ X_j{}^{(k,t)} - W_{ij}(n,t) \qquad (N/2 < j \le N) \end{cases} \qquad (2)$$

で与えられる．ここで，β $(0.5 < \beta < 1)$ は重み付け係数である．

(3) $\| \boldsymbol{y_i}(n,t) \|$ が最小となるマップ層のニューロンを勝ちニューロン r とする．

Learning of Analog Sequential Patterns in Kohonen Feature Map Associative Memory using Area Representation
Tomonori Shiratori and Yuko Osana (Tokyo University of Technology, osana@cc.teu.ac.jp)

(4) 重みの値が固定されていないニューロンに結合する重みベクトルを以下の式に基づいて更新する．

$$\boldsymbol{W_i}(n,t+1) = \boldsymbol{W_i}(n,t) + H(d_i)\alpha(n)h_{ri}\boldsymbol{y_i}(n,t) \quad (3)$$

ここで，$H(d_i)$ は

$$H(d_i) = \frac{1}{1+\exp(-\frac{d_i-D}{\varepsilon})} \quad (4)$$

で与えられる．

(5) $t = t_k - 1$ になるまで (2)~(4) を繰り返す．$t = t_k - 1$ になったら差分ベクトルを $\boldsymbol{y_i} = \boldsymbol{0}$ として初期化する．なお，重みベクトルは以下のように次の学習の繰り返しに引き継がれる．

$$\boldsymbol{W_i}(n+1,1) = \boldsymbol{W_i}(n,t_k) \quad (5)$$

(6) $n = n_{max}$ になるまで (2)~(5) を繰り返す．最後の繰り返しで選ばれた $t_k - 1$ 個の勝ちニューロンの重みを固定する．

(7) (2)~(6) をすべての k に対して繰り返す．

この学習アルゴリズムでは，あるパターンを表現するように十分に学習が行われたマップ層のニューロンに結合する重みを固定，その周辺の重みを準固定することによって，以前に学習した時系列パターンの情報を破壊することなく，新しい時系列パターンを記憶することができる．

2.3 想起過程

想起過程では，入出力層の左側の部分にパターンが入力されると対応するパターンは入出力層の右側の部分に出力される．

パターン $\boldsymbol{X}$ が入力されたときのマップ層のニューロン i の出力 $x_i^{map}(t)$ は

$$x_i^{map}(t) = \begin{cases} 1, & \text{if } \| \boldsymbol{y_i}(t) \| < \theta^{map} \\ 0, & \text{それ以外} \end{cases} \quad (6)$$

$$y_{ij}(n,t) = \begin{cases} (1-\beta)y_{ij}(n,t-1) & \\ \quad +\beta(X_j{}^{(k,t)} - W_{ij}(n,t)) & (1 \le j \le N/2) \\ 0 & (N/2 < j \le N) \end{cases} \quad (7)$$

で与えられる．ここで，θ^{map} はマップ層ニューロンのしきい値である．また，入出力層のニューロン k の出力 $x_k^{in}(t)$ は

$$x_j^{in}(t) = \frac{1}{\sum_i x_i^{map}} \sum_{i:x_i=1} W_{ij}. \quad (8)$$

で与えられる．

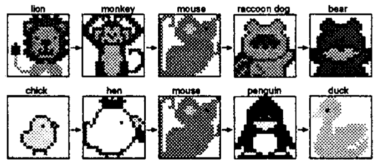
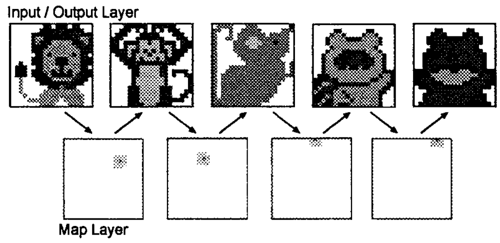
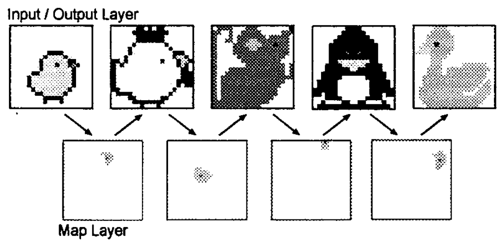


図 2: 学習させる時系列パターンの例



(a) lion



(b) cat

図 3: 想起結果

3 計算機実験

提案モデルにおいて，図 2 のような 2 つの時系列パターンを逐次学習させたときに，*lion* と *chick* の 2 つのパターンを入力層に与えたときの想起結果を図 3 に示す．図 3 より，提案モデルにおいて共通項を含むアナログ時系列パターンの想起が行えることが分かる．

参考文献

[1] N. Sakurai, M. Hattori and H. Ito : "SOM Associative Memory for Temporal Sequences," Proc. of IEEE and INNS International Joint Conference on Neural Networks, pp.950-955, Honolulu, 2002.

[2] 池田成宏, 萩原将文: "新しい知識表現法の提案とニューラルネットワークによる実現," 電子情報通信学会論文誌, Vol.J81-D-II, No.6, pp.1328–1335, 1998.

[3] H. Ichiki, M. Hagiwara and M. Nakagawa: "Kohonen feature maps as a supervised learning machine," Proc. of IEEE International Conference on Neural Networks, pp.1944–1948, 1993.

[4] Y. Iwai and Y. Osana : "Kohonen feature map associative memory with area representation for sequential patterns," Proceedings of International Symposium on Nonlinear Theory and its Applications, Vancouver, 2007.