

高次 Elman 型ニューラルネットワークによる 有限オートマトン同定問題の近似解法

田中 賢[†] 河西 大海^{††}

未知の正則言語の正例と負例の集合が与えられたとき、これと無矛盾な N 状態以下の有限オートマトンを求める問題は、有限オートマトン同定問題とよばれ NP-完全である。ニューラルネットワークの学習能力に基づき、有限オートマトン同定問題の近似解を求める場合、定められた状態数以下の有限オートマトンのみが解となるような学習モデルが必要となる。また、解の存在を保証するためには、初期状態の最終状態集合への所属を学習時に決定できる仕組みが必要となる。本論文では、 $\lceil \log_2 N \rceil$ 次元空間内の点 $\{(p_1, p_2, \dots, p_{\lceil \log_2 N \rceil}) : p_i \in \{0, 1\}\}$ の近傍にニューラルネットワークの状態を集約し、結合重みに制約を加えて同値な状態集合を構成することで、状態数制約を満足できる近似解法を提案する。初期状態の所属を決定するために、所属を判別する出力ユニットを備える高次 Elman 型ニューラルネットワークを構成する。勾配法に基づく学習アルゴリズムを用いて初期状態の所属を学習により決定することを可能とする。富田文法とランダム生成文法を例とし、この学習モデルが状態数制約を満たす有限オートマトンを 11% から 93% の割合で同定できることを計算機実験により確認する。

An Approximate Method for Finite Automata Identification Problem via Higher-Order Elman Neural Networks

KEN TANAKA[†] and OHMI KAWANISHI^{††}

Identification of an automaton with N states which agrees with a finite set of positive or negative examples is NP-complete. In order to get a solution approximately via learning method for neural networks, learning models which can realize any finite automata within N states are needed. In addition, the learning models need to determine the assignment of initial state for final states set. In this paper, we propose a approximate method which can satisfy the limitation of states. The neighbourhoods of $\{(p_1, p_2, \dots, p_{\lceil \log_2 N \rceil}) : p_i \in \{0, 1\}\}$ vertices in $\lceil \log_2 N \rceil$ dimensional space are regarded as gathering points. We propose Higher-Order Elman Neural Network with output unit which assigns the initial state. We derive a learning algorithm which can determine the assignment by use of gradient method. We show that our learning model can identify the automata in a ratio from 11% to 93% within provided number of states by computer simulation.

1. はじめに

パターン処理や信号処理などの非記号的情報処理にとどまらず、言語処理や推論システムなど記号的情報処理に対してもニューラルネットワークの学習汎化能力を生かす試みが近年報告されている^{1),9),13)}。

再帰型ニューラルネットワークは、チューリング機械の計算能力を包含する^{8),16)}。形式言語の語を符号化して時系列的にニューラルネットワークに入力し、

これに従ってニューロンの状態更新を行えば、ニューラルネットワークの最終状態によって語の認識を行うことができる。最終状態に対する教師信号を与え、これと最終状態との差が最少となるような重みを学習することで、入力記号列に無矛盾な有限オートマトンをニューラルネットワーク上に獲得することができる^{2),4),5),10),12),14)}。

有限オートマトン同定問題は、未知の正則言語の正例と負例が与えられたとき、これと無矛盾な有限オートマトンを求める問題で、NP-完全である⁶⁾。同定問題の難しさは、制限された状態数 N の中で例題に無矛盾な状態遷移関数を求める遷移割当て問題が充足問題⁷⁾へ帰着される点にある。状態数の制約を考慮しな

[†] 新潟大学工学部情報工学科
Faculty of Engineering, Niigata University
^{††} 日本電信電話株式会社
Nippon Telegraph and Telephone Corporation

い場合には、自明な解が存在し同定問題の本質的な難しさが失われる。

ニューラルネットワークによる有限オートマトンの学習法は、同定問題の近似解法として用いることができるが、適用にあたっては定められた状態数以下の有限オートマトンのみが解となるような学習モデルが必要となる。Cleeremansら²⁾は、Elman型ニューラルネットワーク³⁾に正則言語の正例のみを符号化して与え、隠れユニットの出力に有限オートマトンの状態が獲得できることを確認したが、負例の存在や状態数の制限については考慮していない。猪飼ら¹⁵⁾は、線形空間オートマトンと等価な2次神経回路網に勾配法を適用することで、オートマトンの状態遷移関数を陽に抽出できることを示した。ここでは、 N 状態有限オートマトンを一意に特定するのに必要な長さ $N-1$ までのすべての語を学習に用いているが、例題として十分な語が与えられない場合の獲得の可否や状態数制限との関連については明らかにしていない。

Gilesら⁵⁾は、入力ユニットと状態ユニットの間に2次の結合構造を持つ再帰型ニューラルネットワークを用いた有限オートマトンの学習法を提案した。Watrousら¹²⁾、Forcadaら⁴⁾、Tiñoら¹⁰⁾、筆者ら¹⁴⁾は、いずれも高次結合を含む再帰型ニューラルネットワークを用いた有限オートマトン獲得率の改善法を示したが、状態数制限については考慮していない。

本論文では、文献14)で提案した再帰型高次結合ニューラルネットワークRHONをもとに状態数制約を満足できる近似解法を提案する。Gilesら⁵⁾、筆者ら¹⁴⁾の学習モデルでは、有限オートマトンの状態を表す状態ユニットの1つを語の判定に用いている。このため、学習にあたっては初期状態が最終状態集合に属するか否かをあらかじめ仮定する必要がある。ここでは、初期状態の所属を動的に決定するために、所属を判別する出力ユニットを備える高次Elman型ニューラルネットワークを新たに提案する。出力ユニットに対して教師信号を与え、出力ユニットとの二乗誤差を評価関数としてこれに勾配法を適用する。評価関数の値に応じてシグモイド関数のパラメータを変化させ、ニューラルネットワークの状態を $\{(p_1, p_2, \dots, p_{\lfloor \log_2 N \rfloor}) : p_i \in \{0, 1\}\}$ の近傍に集約する^{*}。学習の際、結合重みに制約を加えて同値な状態集合を構成する。これにより、定められた状態数の中での状態の集約と初期状態の所属の決定を同時に実現する。

本論文の構成は以下のとおりである。2章では、高

次Elman型ニューラルネットワークを構成し、有限オートマトンの実現法を概説する。3章では、勾配法に基づく学習アルゴリズムを導出し、状態制約の導入法と状態集約のための勾配の制御法を示す。4章では、富田文法とランダム生成有限オートマトンを例とした計算機実験を行い、高次Elman型ニューラルネットワークが状態数制約を満たす有限オートマトンを11%から93%の成功率で同定できることを示す。

本論文で提案した手法は、ニューラルネットワークによる正則言語の学習を有限オートマトン同定問題の近似解法として明確に位置づける。また、ニューラルネットワークの学習法が、言語学習や記号処理の情報処理において発生する困難な問題に対して、有効な近似解を与えられることを示している。

2. 高次Elman型ニューラルネットワークによる有限オートマトンの実現

本章では、高次Elman型ニューラルネットワークの構造と動作を概説する。

2.1 高次Elman型ニューラルネットワークの構成

高次Elman型ニューラルネットワークHEN(Higher-Order Elman Network)は、図1に示すような、入力ユニット、状態ユニット、出力ユニットからなる再帰型ニューラルネットワークモデルである。図中、 m 個の入力ユニットは異なる m 個の入力記号にそれぞれ対応する。各時刻では、その時刻に入力される入力記号に対応する入力ユニットだけが1を出力し、他の入力ユニットは0を出力する。離散化された各時刻 t における第 k 入力ユニットの出力を $x_k^{(t)}$ で表す。 n 個の状態ユニットの出力により有限オートマトンの状態を、出力ユニットの出力により語の所属を表す。第 i 状態ユニットの出力を $O_i^{(t)}$ で、出力ユニットの出力を $H^{(t)}$ で表す。

状態更新のための高次結合は、 n 個の状態ユニットのすべての部分集合と入力ユニットから、すべての状態ユニットの間に存在する。状態の更新はこれら2次から $n+1$ 次までのすべての高次結合を計算することで進行する。 $2 \leq p \leq n+1$ とし、状態ユニット j_1, j_2, \dots, j_{p-1} と入力ユニット k から状態ユニット i への p 次の結合を $w_{j_1 j_2 \dots j_{p-1} k}$ と表す。

所属判別のための高次結合は、 n 個の状態ユニットのすべての部分集合から出力ユニットの間に存在する。 $1 \leq p \leq n$ とし、状態ユニット j_1, j_2, \dots, j_p から出力ユニットへの p 次の結合を $v_{j_1 j_2 \dots j_p}$ と表す。図中の小さい黒丸は高次の結合構造を表すものとする。このニューラルネットワークの状態ユニット i の時刻 $t+1$

^{*}「 x 」は、 x を超える最小の整数とする。

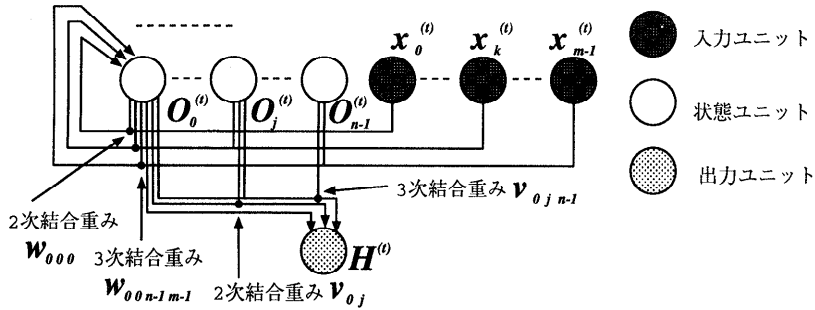


図1 高次 Elman 型ニューラルネットワークの構造
Fig.1 The structure of Higher-Order Elman Network.

における状態 $I_i^{(t+1)}$ を次のように定義する。

$$\begin{aligned}
 I_i^{(t+1)} = & \sum_{j_1=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1k} O_{j_1}^{(t)} x_k^{(t)} \\
 & + \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1j_2k} O_{j_1}^{(t)} O_{j_2}^{(t)} x_k^{(t)} \\
 & \quad (j_1 < j_2) \\
 & + \dots \\
 & + \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} \dots \sum_{j_n=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1j_2 \dots j_n k} \\
 & \quad (j_1 < j_2 < \dots < j_n) \\
 & \quad O_{j_1}^{(t)} O_{j_2}^{(t)} \dots O_{j_n}^{(t)} x_k^{(t)} \quad (1)
 \end{aligned}$$

時刻 $t+1$ における状態ユニット i の出力 $O_i^{(t+1)}$ を、次のように定義する。

$$O_i^{(t+1)} = g(I_i^{(t+1)}) \quad (2)$$

ここで g は状態ユニット i の入出力関数を表す。 g として、以下のようなシグモイド関数、また閾値関数を用いることができる。

$$g(x) = \frac{1}{1 + e^{-\frac{(x+\theta)}{\tau}}} \quad (3)$$

このニューラルネットワークの出力ユニットの時刻 $t+1$ における出力 $H^{(t+1)}$ を次のように定義する。

$$\begin{aligned}
 H^{(t+1)} & = g(V^{(t+1)}) \\
 & = g\left(\sum_{j_1=0}^{n-1} v_{j_1} O_{j_1}^{(t)} + \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} v_{j_1j_2} O_{j_1}^{(t)} O_{j_2}^{(t)} + \dots \right. \\
 & \quad \left. + \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} \dots \sum_{j_n=0}^{n-1} v_{j_1j_2 \dots j_n} O_{j_1}^{(t)} O_{j_2}^{(t)} \dots O_{j_n}^{(t)}\right) \quad (4)
 \end{aligned}$$

HEN はたかだか $n+1$ 次の高次結合を含む。 HEN

に含まれる高次結合の最大次数が p であるとき、HEN の次数は p であるといい、この HEN を p 次の HEN とよぶ。

HEN は、文献 14) で提案した RHON に所属判別のための出力ユニットを付加したニューラルネットワークモデルである。 閾値関数を入出力関数とした $n+1$ 次 HEN は、RHON と同様に n 個の状態ユニット出力において任意の写像 $f: \{0,1\}^n \rightarrow \{0,1\}^n$ を実現できる。 このことは、HEN が任意の状態遷移関係を実現できることに相当する。 これより、HEN は解となる有限オートマトンをつねに探索空間内に含む。 HEN により実現可能な有限オートマトンの最大状態数は、 $\{0,1\}^n$ から原点を除いた点の数となり、 $2^n - 1$ となる。

3. 有限オートマトン同定のための学習アルゴリズム

本章では、入出力関数が微分可能なシグモイド関数の場合について、勾配法を用いた HEN の学習アルゴリズムを導出する。

3.1 勾配法によるオンライン学習アルゴリズム

出力ユニットの最終出力を H^f 、このときの出力ユニットの状態を V^f 、各状態ユニットの出力を $O_{j_1}^f, O_{j_2}^f, \dots, O_{j_p}^f$ と表記する。 学習は、 H^f に対する教師信号を用いて行う。 正例が入力されたとき、 $H^f \approx 1$ 、負例が入力されたとき $H^f \approx 0$ となる学習評価関数を、 H^f と教師信号 H^d の 2 乗誤差によって次のように定義する。

$$E = \frac{1}{2} (H^f - H^d)^2 \quad (5)$$

k 回学習を行ったときの結合重みを $w_{ij_1 \dots j_l k}^{(k)}$ 、 $v_{j_1 \dots j_l}^{(k)}$ 、学習による結合重みの変化分を $\Delta w_{ij_1 \dots j_l k}^{(k)}$ 、 $\Delta v_{j_1 \dots j_l}^{(k)}$ とする。 $k+1$ 回学習後の結合重みを次のように定義する。

$$\begin{aligned} w_{ij_1 \dots j_l k}^{(k+1)} &= w_{ij_1 \dots j_l k}^{(k)} + \Delta w_{ij_1 \dots j_l k}^{(k)} \\ v_{j_1 \dots j_l}^{(k+1)} &= v_{j_1 \dots j_l}^{(k)} + \Delta v_{j_1 \dots j_l}^{(k)} \end{aligned} \quad (6)$$

η を任意の正の実数として、 $\Delta w_{ij_1 \dots j_l k}^{(k)}$ 、 $\Delta v_{j_1 \dots j_l}^{(k)}$ を次のように定義する。

$$\begin{aligned} \Delta w_{ij_1 \dots j_l k}^{(k+1)} &= -\eta \frac{\partial E}{\partial w_{ij_1 \dots j_l k}^{(k)}} \\ \Delta v_{j_1 \dots j_l}^{(k+1)} &= -\eta \frac{\partial E}{\partial v_{j_1 \dots j_l}^{(k)}} \end{aligned} \quad (7)$$

式(5)および(7)より、

$$\begin{aligned} \frac{\partial E}{\partial w_{ij_1 \dots j_l k}^{(k)}} &= (H^f - H^d) \frac{\partial H^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \\ \frac{\partial E}{\partial v_{j_1 \dots j_l}^{(k)}} &= (H^f - H^d) \frac{\partial H^f}{\partial v_{j_1 \dots j_l}^{(k)}} \end{aligned} \quad (8)$$

となる。 $\frac{\partial H^f}{\partial w_{ij_1 \dots j_l k}^{(k)}}$ は、以下の式に従って再帰的に求めることができる。

$$\begin{aligned} & \frac{\partial H^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \\ &= H^f (1 - H^f) \left\{ \sum_{j_1=0}^{n-1} v_{j_1} \frac{\partial O_{j_1}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \right. \\ &+ \sum_{\substack{j_1=0 \\ (j_1 < j_2)}}^{n-1} \sum_{j_2=0}^{n-1} v_{j_1 j_2} \left(\frac{\partial O_{j_1}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} O_{j_2}^f \right. \\ &+ \left. \left. O_{j_1}^f \frac{\partial O_{j_2}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \right) \right. \\ &+ \dots + \sum_{\substack{j_1=0 \\ (j_1 < j_2 < \dots < j_n)}}^{n-1} \sum_{j_2=0}^{n-1} \dots \sum_{j_n=0}^{n-1} v_{j_1 j_2 \dots j_n} \\ &\left. \left(\frac{\partial O_{j_1}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} O_{j_2}^f \dots O_{j_n}^f \right. \right. \\ &+ \left. \left. O_{j_1}^f \frac{\partial O_{j_2}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \dots O_{j_n}^f \right. \right. \\ &+ \dots \\ &+ \left. \left. O_{j_1}^f O_{j_2}^f \dots \frac{\partial O_{j_n}^f}{\partial w_{ij_1 \dots j_l k}^{(k)}} \right) \right\} \end{aligned} \quad (9)$$

$$\begin{aligned} & \frac{\partial O_i^{(t+1)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} \\ &= O_i^{(t+1)} (1 - O_i^{(t+1)}) \left\{ O_{j_1}^{(t)} \dots O_{j_l}^{(t)} x_k^{(t)} \right. \\ &+ \sum_{j_1=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1 k} \frac{\partial O_{j_1}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} x_k^{(t)} \\ &+ \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1 j_2 k} \left(\frac{\partial O_{j_1}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} O_{j_2}^{(t)} \right. \\ &+ \left. \left. O_{j_1}^{(t)} \frac{\partial O_{j_2}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} \right) x_k^{(t)} + \dots \right. \\ &+ \sum_{j_1=0}^{n-1} \sum_{j_2=0}^{n-1} \dots \sum_{j_n=0}^{n-1} \sum_{k=0}^{m-1} w_{ij_1 j_2 \dots j_n k} \\ &\left. \left(\frac{\partial O_{j_1}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} O_{j_2}^{(t)} \dots O_{j_n}^{(t)} \right. \right. \\ &+ \left. \left. O_{j_1}^{(t)} \frac{\partial O_{j_2}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} \dots O_{j_n}^{(t)} + \dots \right. \right. \\ &+ \left. \left. O_{j_1}^{(t)} O_{j_2}^{(t)} \dots \frac{\partial O_{j_n}^{(t)}}{\partial w_{ij_1 \dots j_l k}^{(t)}} \right) x_k^{(t)} \right\} \end{aligned} \quad (10)$$

$\frac{\partial H^f}{\partial v_{j_1 \dots j_l}^{(k)}}$ は、出力ユニットからの誤差信号により、以下の式に従って直接求めることができる。

$$\begin{aligned} \frac{\partial H^f}{\partial v_{j_1 \dots j_l}^{(k)}} &= \frac{\partial g(V^f)}{\partial v_{j_1 \dots j_l}^{(k)}} \\ &= \frac{dg(V^f)}{dV^f} \frac{\partial V^f}{\partial v_{j_1 \dots j_l}^{(k)}} \\ &= H^f (1 - H^f) \{ O_{j_1}^f \dots O_{j_l}^f \} \end{aligned} \quad (11)$$

ここで、 $O_i^{(0)}$ は $w_{ij_1 \dots j_l k}$ および $v_{j_1 \dots j_l}$ によらず一定のため、 $\frac{\partial O_i^{(0)}}{\partial w_{ij_1 \dots j_l k}} = 0$ 、 $\frac{\partial O_i^{(0)}}{\partial v_{j_1 \dots j_l}} = 0$ とおくことができる。以上により、例題の提示と並行して $\frac{\partial O_i^{(t)}}{\partial w_{ij_1 \dots j_l k}}$ 、 $\frac{\partial O_i^{(t)}}{\partial v_{j_1 \dots j_l}}$ の計算を行うオンライン学習が実現できる。実際の学習では、すべての例題について式(7)の計算を行い、変化の平均値を求めたうえで一括して結合重みを更新する。

HENでは g が閾値関数のとき任意の有限オートマトンを実現できるが、学習を行うには g が微分可能であることが必要となる。ここでは、シグモイド関数により閾値関数と同様の有限な状態空間を実現するために、 $\{0, 1\}$ を座標値とする点が安定不動点となるようシグモイド関数のパラメータ T を十分小さな値とする。 $T \approx 0$ のときには、重みの更新値が著しく小さく、もしくは大きくなり学習が不安定となる。そこで、 E の値に応じて T を変化させる。 α を任意の実数として、以下に従い T を変化させることにする。

$$T = \alpha E \quad (12)$$

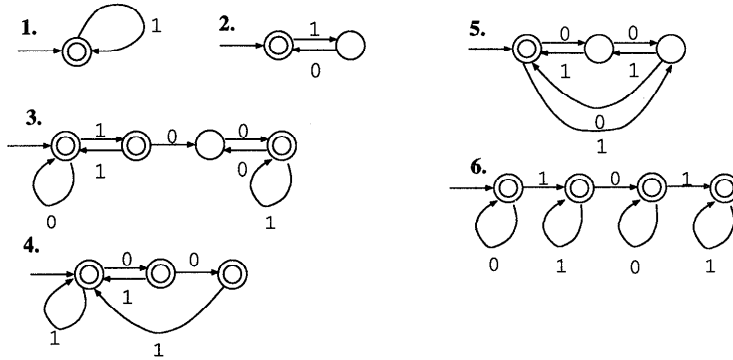


図2 富田文法の解となる最小有限オートマトン
Fig. 2 The minimum automata of Tomita's Grammar.

HENは最大次数までの高次の結合重みを含むため、閾値関数のバイアス θ に依存せず任意の状態遷移関係が実現可能である。ここでは θ を 0 とし、学習による値の更新は行わないこととする。

3.2 解の存在と HEN への状態数制約の導入法

HENでは、 n 次元状態ベクトルの集合により有限オートマトンの状態集合を表す。学習開始時における有限オートマトンの状態ベクトル集合は、ランダムに与えられた初期結合重みに依存する。 $n + 1$ 次 HEN は、 n 個の状態ユニット出力から n 個の状態ユニット出力への任意の写像 $f: \{0, 1\}^n \rightarrow \{0, 1\}^n$ を実現できる。このため、学習開始時の状態ベクトル集合によらず、HENは解となる有限オートマトンをつねに探索空間内に含む。

$n = 1, 2, 3, 4, \dots$ の HEN は、それぞれたかだか状態数 $1, 3, 7, 15, \dots$ の任意の有限オートマトンを実現できる。このことを利用して、 n 次 HEN を同定問題の近似解法に適用できる。任意の正整数 N を状態数制約とする同定問題に HEN を適用するには、 $\lceil \log_2 N \rceil$ の状態ユニットからなる HEN の状態数が N となるように、HEN が状態としてとりうる状態ベクトルをあらかじめ削減すればよい。このために、状態ベクトル集合の中に同値となる状態ベクトル集合が存在するように、結合重みの値に対し学習時に制約を与える方法を考える。

k 番目の入力記号を受けたとき、状態ユニット i へ向かう任意の 2 つの異なる結合重み $w_{is_1s_2 \dots s_p k}$, $w_{it_1t_2 \dots t_q k}$ を考える。状態ユニット $s_1s_2 \dots s_p$ のみが 1 を出力する状態を q_1 , $t_1t_2 \dots t_q$ のみが同時に 1 を出力する状態を q_2 とすると、2 つの結合重みは状態 q_1 , q_2 からの遷移を行う場合のみ状態ユニット i の出力を 0, 1 いずれにも決定できる能力を持つ。学習にあたり、 $w_{is_1s_2 \dots s_p k} = w_{it_1t_2 \dots t_q k}$ なる制約を

$i = 0, 1, \dots, n - 1$ すべての結合重みに対して与えることで、 q_1 と q_2 は k 番目の入力記号を受けたときつねに同じ状態に遷移することとなる。すべての入力記号について、同様に同じ状態に遷移するよう結合重みを等しく設定することで、 q_1 と q_2 を状態ベクトル集合の中の同値な状態ベクトルとすることができる。以上の操作を、 $2^{\lceil \log_2 N \rceil}$ の状態ベクトル集合内に $2^n - 1 - N$ 組行うことで、任意の正整数 N を状態数制約とする同定問題に適用できる HEN を構成することができる。

4. 有限オートマトン同定実験

本章では、HEN の学習能力を計算機実験により確認する。

4.1 状態数制約の効果

Tomita¹¹⁾によって提案され、Giles^ら⁵⁾, Watrous^ら¹²⁾などの報告で実験に用いられた富田文法を例題としてとりあげる。ここでは、以下の 6 種類の富田文法を用いる*。

- (1) 1^*
- (2) $(10)^*$
- (3) $(0 + 11)^* \{ \epsilon + 1 + 100(00 + 1)^* \}$
- (4) $(1 + 01 + 001)^* (\epsilon + 0 + 00)$
- (5) $(\epsilon + 01 + 10 + 000 + 111)^*$
- (6) $0^* 1^* 0^* 1^*$

解となる有限オートマトンを図 2 に示す。

重みを固定しない場合の状態数制約の効果を確かめる。状態ユニット数を 3, 2, 1 とし、それぞれの問題について長さ 10 までの正例と負例をそれぞれ 10 例ずつ 20 例ランダムに選択した。 $\eta = 0.5$, $\alpha = 3.0$ とし

* 文献 11) で提案された 7 文法のうち、5 番目の文法については複数の解釈が存在するため、実験から除外した。

表1 富田文法の同定実験の結果
Table 1 The results of identification for Tomita's Grammar.

文法	状態数	平均収束値	成功率	振動率	平均更新回数	1	2	3	4	5	6	7
富田 1	1	0.0256	85	3	35	32	38	12	3	0	0	0
富田 2	2	0.0324	47	2	119	0	21	13	6	5	2	0
富田 3	4	0.0532	14	0	182	0	0	(3)	11	0	0	0
富田 4	3	0.0409	14	0	85	0	0	7	5	1	0	1
富田 5	3	0.0732	17	2	301	0	0	8	6	3	0	0
富田 6	4	0.0697	36	0	215	0	0	0	19	13	4	0

表2 最大状態数3の場合の同定実験(状態ユニット数2)
Table 2 The identification results in 3 states finite automata (2 state units).

文法	状態数	成功率	1	2	3	平均更新回数
富田 1	1	91	29	42	20	28
富田 2	2	76	0	43	26	7
富田 3	4	0	0	0	0	-
富田 4	3	19	0	0	19	57
富田 5	3	28	0	0	28	224
富田 6	4	0	0	0	0	-

表3 最大状態数1の場合の同定実験(状態ユニット数1)
Table 3 The identification results in 1 states finite automata (1 state units).

文法	状態数	成功率	1	平均更新回数
富田 1	1	93	82	15
富田 2	2	0	0	-
富田 3	4	0	0	-
富田 4	3	0	0	-
富田 5	3	0	0	-
富田 6	4	0	0	-

て、100通りの異なった初期重みから学習を行い、結合重みの更新が1000回以内で $E < 0.01$ となった場合に学習に成功したものとした。同定された有限オートマトンの数を、各状態数ごとに分類した結果を表1, 表2, 表3に示す。表中、状態数は同定の対象となるオートマトンの状態数、平均収束値は学習終了時の評価関数の平均値、成功率は無矛盾なオートマトンの数、振動率は学習が振動した割合、平均更新回数は成功例についての更新回数の平均値を示す。

状態ユニット数3の場合、すべての問題において無矛盾な有限オートマトンが獲得できた。富田3文法については、最小オートマトンより状態数が少ない4状態オートマトンが獲得されている。これは、選択した例題に無矛盾となるより小さな有限オートマトンが存在したことによる。状態ユニット数2の場合、実現可能なオートマトンの最大状態数は3となる。無矛盾オートマトンの状態数が3を超える富田3文法、富田6文法については、無矛盾オートマトンが得られなかった。状態ユニット数1の場合にも同様の結果を得た。これらのことから、勾配法により適切な状態遷移関数が学習できること、重みを固定しない場合の状態数制約が効果的に働くことが確認できた。

4.2 重みの固定による状態数制約の効果

3.2節の方法により重みを固定した場合の状態数制約の効果を確認する。状態ユニット数を3, 2とし、最大状態数がそれぞれ5, 2となるよう重みを固定した。4.1節と同様の条件のもとで行った実験の結果を表4, 表5に示す。

すべての問題において無矛盾な有限オートマトンが

獲得できた。結合重みに制約を加えない場合と比較して、状態数の多い有限オートマトンの獲得率が高くなっている。これは、重みを固定したことにより探索空間が小さくなり探索の効率が向上したことによる。状態ユニット数3, 2いずれの場合も、最大状態数を超える有限オートマトンは獲得されなかった。このことより、結合重みの制約が有限オートマトンの状態数制約に対し適切に機能していることが確認できた。

4.3 初期状態割当ての効果

富田文法で同定対象となる有限オートマトンは、初期状態が最終状態集合に属するもののみである。学習による初期状態割当ての効果を確認するため、初期状態が最終状態集合に属する有限オートマトン、属さない有限オートマトンをそれぞれ3つずつランダムに生成し、状態ユニット数3, 最大状態数5として同定実験を行った。実験の結果を表6に示す。

所属型、非所属型双方の例題で無矛盾な有限オートマトンが獲得できた。ランダムに生成された有限オートマトンは、富田文法と等価な有限オートマトンと比較してより多くのアークを含む傾向がある。このため、富田文法と比較してより多くの平均更新回数を要している。学習の振動率も高くなる傾向があるのも同様の理由による。

5. 結 論

従来、ニューラルネットワークによる正則言語の学習に関する報告では、例題に合致する有限オートマトンを連続空間内で同定し、状態集合のクラスタリングや遷移関数の抽出を行うことを目的としており、状態

表 4 最大状態数 5 の場合の同定実験 (状態ユニット数 3)
Table 4 The identification results in 5 states finite automata (3 state units).

文法	状態数	平均収束値	成功率	振動率	平均更新回数	1	2	3	4	5
富田 1	1	0.0182	67	1	38	28	30	5	4	0
富田 2	2	0.0243	39	5	187	0	18	10	8	3
富田 3	4	0.0725	20	0	83	0	0	(3)	15	2
富田 4	3	0.0601	28	0	87	0	0	16	7	5
富田 5	3	0.0744	21	1	332	0	0	11	8	2
富田 6	4	0.0807	19	0	202	0	0	0	9	10

表 5 最大状態数 2 の場合の同定実験 (状態ユニット数 2)
Table 5 The identification results in 2 states finite automata (2 state units).

文法	状態数	成功率	1	2	平均更新回数
富田 1	1	73	24	37	12
富田 2	2	59	0	22	37
富田 3	4	1	0	1	(1)
富田 4	3	22	0	0	22
富田 5	3	17	0	0	17
富田 6	4	0	0	0	0

表 6 初期状態割当ての効果 (状態ユニット数 3)
Table 6 The effect of initial state assignment (3 state units).

文法	状態数	平均収束値	成功率	振動率	平均更新回数	1	2	3	4	5
所属型 1	3	0.0132	57	9	263	0	0	32	10	15
所属型 2	4	0.0481	21	16	344	0	0	0	17	4
所属型 3	5	0.1635	11	2	923	0	0	0	6	5
非所属型 1	3	0.0992	48	10	320	0	0	22	15	11
非所属型 2	3	0.0201	29	2	392	0	0	13	12	4
非所属型 3	4	0.0771	14	8	472	0	0	0	12	2

数制約と同定問題の関連については明らかにされてこなかった。本論文では、ニューラルネットワークによる正則言語の学習を有限オートマトン同定問題の近似解法として明確に位置づけるため、状態数制約を導入できるニューラルネットワークモデル HEN を提案した。勾配法に基づき、初期状態の所属を学習により決定できる学習アルゴリズムを導出した。富田文法とランダム生成文法を例とし、HEN が状態数制約を満たす有限オートマトンを 11% から 93% の割合で同定できることを計算機実験により確認した。これらの結果は、ニューラルネットワークの学習法が、言語学習や記号処理的情報処理における NP-完全な問題に対して、有効な近似解を与える可能性があることを示している。

本論文では、シグモイド関数の勾配を E の値より制御することで、有限オートマトンの状態を頂点近傍の安定不動点に集約する経験的な手法を用いている。 T の値が 0 に近づくと、 g の微分値が急激に増加する点が存在し学習が不安定となる。今後、学習の振動を抑制しながら状態の集約を同時に実現できる最適な T の制御法を明らかにする必要がある。

計算機実験では、同定対象となる有限オートマトン

の状態数が多くなるにつれ学習成功率が低下している。これは、状態数の増加にともなって、探索空間に含まれる解の割合が変化すること、また遷移関係の複雑化により、勾配による結合重みの更新が必要な遷移関係を実現できなくなることが一因である。今後、これら問題を解決する学習アルゴリズムを開発する必要がある。

本論文では、正則言語の受理系である有限オートマトンを対象とした同定問題の近似解法を提案した。一般に、言語の学習は有限長で有限個の例題集合のもとで行われるため、有限オートマトンより上位の受理系の同定問題は意味をなさない。一方、例題に一定の情報を付加することでプッシュダウンオートマトンなどより上位の同定問題を形式化できることが知られている。今後、文脈自由言語を例題とした同定問題を HEN 上で形式化する方法を考案する必要がある。

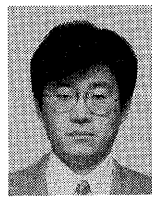
参考文献

- 1) Chan, S.W.K. and Franklin, J.: Symbolic Connectionism in Natural Language Disambiguation, *IEEE Trans. on NN*, Vol.9, No.5, pp.739-755 (1995).

- 2) Cleeremans, A., Servan-Schreiber, D. and McClelland, J.: Finite state automata and simple recurrent networks, *Neural Computation*, Vol.1, No.3, pp.372-381 (1989).
- 3) Elman, J.L.: Finding structure in time, *Cog. Sci.*, Vol.14, pp.179-211 (1990).
- 4) Forcada, M.L. and Carrasco, R.C.: Learning the Initial State of a Second-Order Recurrent Neural Network during Regular-Language Inference, *Neural Computation*, Vol.7, pp.923-930 (1995).
- 5) Giles, C., Miller, C., Chen, D., Chen, H., Sun, G. and Lee, Y.: Learning and extracting finite state automata with second-order recurrent neural networks, *Neural Computation*, Vol.4, No.4, pp.393-405 (1992).
- 6) Gold, E.M.: Complexity of automaton identification from given data, *Inf. & Control*, Vol.37, No.3, pp.302-320 (1978).
- 7) Hopcroft, J.E. and Ullman, J.E.: オートマトン言語理論 計算論 II, サイエンス社 (1984).
- 8) Siegelmann, H.T. and Sontag, E.D.: On the Computational Power of Neural Nets, *Journal of Computer and System Sciences*, Vol.50, pp.132-150 (1995).
- 9) Tiño, P. and Köteles, M.: Extracting Finite-State Representations from Recurrent Neural Networks Trained on Chaotic Symbolic Sequences, *IEEE Trans. on NN*, Vol.10, No.2, pp.284-302 (1999).
- 10) Tiño, P. and Šajda, J.: Learning and Extracting Initial Mealy Automata with a Modular Neural Network Model, *Neural Computation*, Vol.7, pp.822-844 (1995).
- 11) Tomita, M.: Dynamic construction of finite automata from examples using hill-climbing, *Proc. 4th Ann. Conf. Cogn. Sci. Soc.*, pp.105-108 (1982).
- 12) Watrous, R.L. and Kuhn, G.M.: Induction of Finite-State Languages Using Second-Order Recurrent Networks, *Neural Computation*, Vol.4, pp.406-414 (1995).
- 13) 田中 賢, 熊沢逸夫: 再帰型高次結合ニューラルネットワークによる文脈自由言語の認識, 情報処理学会論文誌, Vol.38, No.5, pp.971-979 (1997).
- 14) 田中 賢, 熊沢逸夫, 小川英光: 再帰型高次結合ニューラルネットワークによる正規言語の学習, 電子情報通信学会論文誌, Vol.J79-D-II, No.5, pp.899-907 (1996).
- 15) 猪飼武夫, 越智貴夫: 2次神経回路網による線形空間および有限状態オートマトンの実現と学習, 信学技報, Vol.NC92-71, pp.17-24 (1992).
- 16) 林原香織, 山下雅史, 阿江 忠: シグモイド関数の連続性/離散性とニューラルネットワークのマシン能力について, 信学論 (DII), Vol.J73-D-II, No.8, pp.1220-1226 (1990).

(平成 11 年 4 月 13 日受付)

(平成 11 年 9 月 2 日採録)



田中 賢 (正会員)

平成元年早稲田大学理工学部電気工学科卒業。平成3年東京工業大学大学院総合理工・システム科学修士課程修了。平成7年同大学院理工・情報工学博士課程単位取得。同年新潟大学大工学部情報工学科助手。ニューラルネットワークの研究に従事。



河西 大海

平成11年新潟大学大学院自然研・情報理工学専攻前期課程修了。現在、NTTコミュニケーションズ株式会社に勤務。ニューラルネットワークと形式言語に関する研究に従事。