

# 記号列を入出力とするニューラルネットの学習法

本 木 実<sup>†</sup> 富 浦 洋 一<sup>††</sup> 高 橋 直 人<sup>†††</sup>

本論文では、記号列を入力し記号列を出力する階層型ニューラルネットの学習法を提案する。本論文で考察するモデルは、結合荷重だけでなく、各記号に対応する記号表現ベクトルも学習パラメータとする。この方式により、学習データの性質を反映した記号表現ベクトル（類似の使われ方をする記号の記号表現ベクトルが互いに近いベクトル）を学習することができ、予測能力の向上が期待できる。しかし、目的関数を平均二乗誤差とする通常のモデルでは、目的関数の値を最小にするタスクにとって無意味な解が存在し、出力ベクトルから記号の同定を行うと正解率がきわめて低いという問題がある。そこで本論文では、記号の同定を考慮した目的関数による学習法を提案する。実験により、提案モデルは、学習データの性質を反映した記号表現ベクトルの学習が可能であり、かつ高い正解率を持つことを示す。

## A Learning Method of a Layered Neural Network Whose Inputs and Outputs are Symbol Sequences

MINORU MOTOKI,<sup>†</sup> YOICHI TOMIURA<sup>††</sup> and NAOTO TAKAHASHI<sup>†††</sup>

This paper proposes a learning method of a layered neural network whose inputs and outputs are symbol sequences. The learning parameters of the model we consider here are not only link weights but also symbol representation vectors (SRVs), each of which corresponds to each symbol. SRVs learned by this model can reflect characteristics of the training data, and are expected to lead to high performance in prediction. The conventional learning method whose objective function is mean square error has meaningless solutions for the task, which minimize the objective function. Moreover, the accuracy of the conventional model is very low when symbols are identified based on the output vectors. The learning method we propose has an objective function which is proper for identification of output symbols. Through the computational experiments, we also show that the proposed model acquires SRVs which reflect characteristics of the training data and have high performance.

### 1. はじめに

パターン情報処理分野で取り扱うパターンは、「アナログパターン」と「記号列（デジタル）パターン」とに大別できる。アナログパターンは、音声や画像データが代表的な例であり、記号列パターンは、自然言語、楽譜、DNA の塩基配列、タンパク質のアミノ酸配列等が例である<sup>1)</sup>。

パターン認識や関数近似で優れた性能を示すモデル

の 1 つに階層型ニューラルネットがある。これまで、ニューラルネットで記号列を取り扱う研究がなされてきた<sup>2)~5)</sup> が、ニューラルネットは本来アナログデータを取り扱うため、記号を実ベクトルに対応させる必要がある。異なる記号を単に異なる実ベクトルに対応させるというのも 1 つの解である。しかしたとえば、与えられたタスクにおいて類似した性質を示す 2 つの記号が類似した実ベクトルに対応するように、各記号に実ベクトルを対応付ける方が、一般には、ニューラルネットの性能が高くなると予想される。したがって、各記号にどのような実ベクトルを対応付けるか（表現の問題）が記号列パターンを取り扱うニューラルネットを構成する場合重要となる。

ニューラルネットを用いて言語現象を説明する認知的科学的な研究の中に、Miikkulainen らの研究がある<sup>6)</sup>。彼らは、単語（すなわち記号）を実ベクトル（以下では、記号表現ベクトル）として表現し、深層格解析等のタ

<sup>†</sup> 九州大学大学院システム情報科学府  
Graduate School of Information Science and Electrical Engineering, Kyushu University

<sup>††</sup> 九州大学大学院システム情報科学研究所  
Faculty of Information Science and Electrical Engineering, Kyushu University

<sup>†††</sup> 独立行政法人産業技術総合研究所  
National Institute of Advanced Industrial Science and Technology (AIST)

スクを担う階層型ニューラルネットのモジュールを多段に連結することで、この記号表現ベクトルがモジュール間を移動するモデルを提案した。このモジュールでは、結合荷重だけでなく記号表現ベクトルも学習パラメータとして学習することが大きな特徴である。この特徴により、学習データの性質を反映した記号表現ベクトル（類似の使われ方をする記号の記号表現ベクトルが互いに近いベクトル）を学習することができるので、上記の表現の問題に1つの解を与えるだけでなく、予測能力の向上も期待できる。このようなニューラルネットを彼らはFGREP (Forming Global Representation with Extended back-Propagation) モジュールと名づけた。

FGREP モジュールの学習法は、BP (Back Propagation) 法の類推から得られている。BP 法是最急降下法に基づいているが、Miikkulainen らの示したパラメータの更新式は最急降下法に従っておらず学習される記号表現ベクトルの数学的な意味が不明確であり、また、出力記号の同定の正解率がきわめて低いという問題がある。さらに、通常のように、目的関数を平均二乗誤差として学習した場合、目的関数の値を最小にするが、与えられたタスクにとって無意味な解が存在する。そこで本論文では、記号列を入出力とする階層型ニューラルネットに対し、記号の同定を考慮した学習法を提案する。

続く2章では、記号列を入出力とするニューラルネットについて述べる。この中で、目的関数を平均二乗誤差とした通常のモデル（最小二乗誤差基準モデル）および提案モデルについて述べる。3章では、関

連研究として FGREP モジュールとその問題について紹介する。4章では、最小二乗誤差基準モデル、記号表現ベクトルを学習しない通常のモデル、提案モデル、FGREP モジュールに関する実験について述べる。5章で、本論文のまとめを行う。

## 2. 記号列を入出力とするニューラルネット

### 2.1 記号列を入出力とするニューラルネットの構成

本論文では、階層型のニューラルネットを用いて記号列から記号列への関数を近似する問題を取り扱う。階層型ニューラルネット自体の入力と出力は実ベクトルで表されるので、入力する記号列を実ベクトルに置き換えて階層型ニューラルネットに入力し、階層型ニューラルネットが出力するベクトルを記号に同定する必要がある。本節では記号列を入出力とするニューラルネットの構成について述べる。

扱う記号の全体集合を  $W = \{W_0, W_1, \dots, W_{V-1}\}$  とする。記号  $W_c$  に対応する  $m$  次元実ベクトルを記号表現ベクトルと呼び、 $r_c = [r_{c,1}, r_{c,2}, \dots, r_{c,m}]^t$  と表す。各記号の記号表現ベクトルは記号表現ベクトル辞書に格納される(図1)。記号表現ベクトル辞書  $L$  は全  $V$  個の記号表現ベクトルを連結した実ベクトル  $L = [r_{0,1}, r_{0,2}, \dots, r_{0,m}, r_{1,1}, \dots, r_{V-1,m-1}, r_{V-1,m}]^t$  である。

ニューラルネットの入力層は、 $S_{in}$  個のユニット(ユニット群)に、出力層は  $S_{out}$  個のユニットに分けられている(ユニット番号は  $0, 1, \dots$  と付与する)。入力層には、 $S_{in}$  個の記号の各記号表現ベクトルを順に連結したベクトルが入力され、出力層には、 $S_{out}$  個

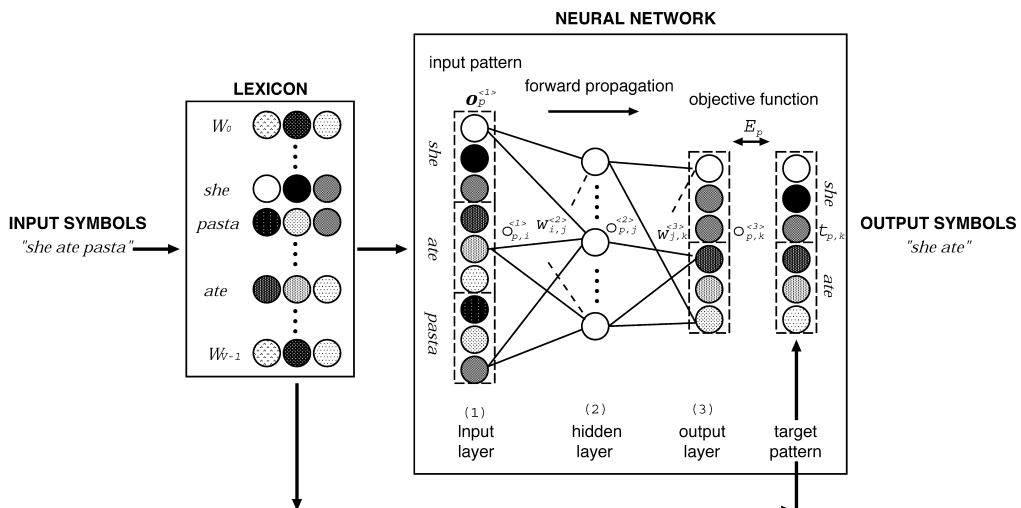


図1 記号列を入出力とするニューラルネットの構成

Fig. 1 Structure of the neural network whose inputs and outputs are symbol sequences.

のベクトルが連結されたベクトルが出力される．各スロットを構成するユニット数は  $m$  (記号表現ベクトルの次元) である．出力ベクトルからの記号列の同定は，ユークリッド距離を用いて行う．すなわち，出力第  $s$  スロットの記号は

$$\arg \min_{w_c \in \mathcal{W}} \frac{1}{m} \sum_{e=1}^m \frac{1}{2} \left( o_{sm+e}^{(3)} - r_{c,e} \right)^2$$

と同定される．

通常の階層型ニューラルネットと同様，入力側から数えて  $l$  番目 ( $l > 1$ ) の層における第  $j$  ユニットの出力値  $o_j^{(l)}$  は，

$$o_j^{(l)} = f(u_j^{(l)}) = f\left(\sum_{i=1}^{n_{l-1}} w_{i,j}^{(l)} o_i^{(l-1)} + b_j^{(l)}\right)$$

で計算される．ここで， $n_l$  は第  $l$  層のユニット数を表し， $w_{i,j}^{(l)}$  は第  $(l-1)$  層第  $i$  ユニットの第  $l$  層第  $j$  ユニットの結合荷重， $b_j^{(l)}$  は第  $l$  層第  $j$  ユニットのバイアスと呼ばれ，これら  $w_{i,j}^{(l)}$ ， $b_j^{(l)}$  は学習パラメータである．全  $w_{i,j}^{(l)}$ ， $b_j^{(l)}$  ( $l = 2, 3; i = 1, 2, \dots, n_{l-1}; j = 1, 2, \dots, n_l$ ) を  $w$  で表し，これも結合荷重と呼ぶことにする． $u_j^{(l)}$  は，第  $l$  層第  $j$  ユニットの内部ポテンシャルと呼ばれる． $f$  としては次のシグモイド関数，

$$f(u) = \frac{1}{1 + \exp(-u)}$$

を用いる．

以上のように，本論文では，記号列から記号列への関数近似の問題を，階層型ニューラルネットを用いて，「入力記号列を記号表現ベクトル辞書  $L$  を用いて実ベクトルに変換し，これを階層型ニューラルネットの入力とし，階層型ニューラルネットが出力する実ベクトルを記号表現ベクトル辞書  $L$  を用いてスロットごとに記号に変換して出力記号列を生成する」という方法で解く．ここで問題となるのが，各記号に対する記号表現ベクトル，つまり，記号表現ベクトル辞書  $L$  をどのように与えるかである．本論文では， $L$  を事前に恣意的に与えるのではなく，入力記号列  $s_p$  とそれに対する望ましい出力記号列 (教師記号列)  $t_p$  の組からなる学習データ  $\langle (s_1, t_1)(s_2, t_2) \dots (s_N, t_N) \rangle$  を用いて，結合荷重だけでなく，記号表現ベクトル辞書  $L$  も同時に学習する手法を採用する．これは最近研究されている単語をベクトルで表すモデル<sup>(6)~(9)</sup>で採用されている方式であり，記号表現ベクトル辞書  $L$  を学習パラメータとすることによって，目的関数を最小とする基準に従って記号表現ベクトルが決定される．学習デー

タ中に類似した使われ方がなされた記号どうしの記号表現ベクトルは互いに近いベクトルとなることが期待でき，類似の未知記号列に対する予測がうまくいくと考えられる．

## 2.2 最小二乗誤差基準モデルとその問題

以降の説明では，教師記号列の各記号を記号表現ベクトルに変換して順に連結したベクトルを教師ベクトルと呼び， $t_p$  に対応する教師ベクトルを  $t_p$  で表す．また， $p$  番目の入力記号列が与えられたときの第  $l$  層，第  $j$  ユニットの出力値を  $o_{p,j}^{(l)}$  と表す．

前節で示した記号列を入力とするニューラルネットを構成しようとするとき，まず，ニューラルネットの出力と教師ベクトルの平均二乗誤差を目的関数として，これを最小化するように学習するモデル (以下 MSE 基準モデル) が考えられる．ところが MSE 基準モデルの場合，「すべての入力ベクトルが同一，かつ，すべての教師ベクトルが同一となる可能性があり，その結果，記号の同定率がきわめて悪くなる」という問題がある．

MSE 基準モデルは，ニューラルネットの出力と教師信号との平均二乗誤差

$$\begin{aligned} E(w, L) &= \frac{1}{N} \sum_{p=1}^N E_p \\ &= \frac{1}{N} \sum_{p=1}^N \sum_{k=1}^{n_3} \frac{1}{2} \left( o_{p,k}^{(3)}(w) - t_{p,k}(L) \right)^2 \end{aligned} \quad (1)$$

を目的関数とし， $E(w, L)$  を最小 (実際には極小) にするように学習パラメータである結合荷重  $w$  と記号表現ベクトル辞書  $L$  を学習する．最急降下法を用いて  $w$  を求めることは，誤差逆伝搬法 (BP 法)<sup>(10)</sup> として一般によく知られている．ここで， $1/2$  は計算の便宜上導入される定数で， $t_{p,k}$  は  $t_p$  の第  $k$  成分である．

たとえば入力ユニットと中間ユニットの結合荷重  $w_{i,j}^{(2)}$  は，

$$w_{i,j}^{(2)} \leftarrow w_{i,j}^{(2)} - \eta \frac{\partial E}{\partial w_{i,j}^{(2)}} \quad (2)$$

$$\frac{\partial E}{\partial w_{i,j}^{(2)}} = \frac{1}{N} \sum_{p=1}^N \delta_{p,j}^{(2)} \cdot o_{p,i}^{(1)} \quad (3)$$

に従って更新される．ただし， $\delta_{p,j}^{(2)}$  は誤差信号と呼ばれ，

$$\delta_{p,j}^{(2)} = f'(u_{p,j}^{(2)}) \cdot \sum_{k=1}^{n_3} \delta_{p,k}^{(3)} \cdot w_{j,k}^{(3)} \quad (4)$$

のように，次段の誤差信号  $\delta_{p,k}^{(3)}$  で表される．

一方、記号表現ベクトル  $r_{c,e}$  の学習は、教師信号が記号表現ベクトルに依存しているため式 (1) 中の  $t_{p,k}$  も  $L$  の関数になっており、

$$r_{c,e} \leftarrow r_{c,e} - \eta \frac{\partial E}{\partial r_{c,e}} \quad (5)$$

$$\begin{aligned} \frac{\partial E}{\partial r_{c,e}} &= \sum_{i=1}^{n_1} \frac{\partial E}{\partial o_{p,i}^{(1)}} \frac{\partial o_{p,i}^{(1)}}{\partial r_{c,e}} + \sum_{k=1}^{n_3} \frac{\partial E}{\partial t_{p,k}} \frac{\partial t_{p,k}}{\partial r_{c,e}} \\ &= \frac{1}{N} \sum_{p=1}^N \sum_{i=1}^{n_1} \delta_{p,i}^{(1)} \cdot [M_p]_{i,cm+e} \\ &\quad + \frac{1}{N} \sum_{p=1}^N \sum_{k=1}^{n_3} \frac{\partial E_p}{\partial t_{p,k}} \cdot [T_p]_{k,cm+e} \end{aligned} \quad (6)$$

$$\delta_{p,i}^{(1)} = \sum_{j=1}^{n_2} \delta_{p,j}^{(2)} \cdot w_{i,j}^{(2)} \quad (7)$$

に従って更新される。ここで  $M_p, T_p$  は、

$$\begin{aligned} [M_p]_{sm+e,cm+e}; \text{ for } e=1,2,\dots,m \\ &= \begin{cases} 1, & p \text{ 番目の入力データの第 } s \\ & \text{スロットの単語が } W_c \text{ の場合} \\ 0, & \text{それ以外の場合} \end{cases} \\ [T_p]_{sm+e,cm+e}; \text{ for } e=1,2,\dots,m \\ &= \begin{cases} 1, & p \text{ 番目の教師データの第 } s \\ & \text{スロットの単語が } W_c \text{ の場合} \\ 0, & \text{それ以外の場合} \end{cases} \end{aligned}$$

であり、これを用いて学習データにおける入力記号列に対応する入力ベクトル  $o_p^{(1)}$ 、教師記号列に対応する教師ベクトル  $t_p$  は、

$$o_p^{(1)} = M_p L, \quad t_p = T_p L$$

と表せる。

ところが、このような学習を行った場合、目的関数  $E$  の値を最小にする自明な、しかも設定したタスクにとって無意味な解が存在する。パラメータ学習は、「目的関数  $E(w, L)$  の値をより小さくする  $w, L$  がより良いパラメータである」という指標に従って行われる。すると、たとえば、

- $\forall p \quad M_p L = x_{in}$  ( $x_{in}$  はある定数ベクトル)

- $\forall p \quad T_p L = x_t$  ( $x_t$  はある定数ベクトル)

- $o^{(1)} = x_{in}$  のとき、 $o^{(3)} = x_t$  となるような  $w$  を満たす ( $w, L$ ) は、 $E(w, L) = 0$  となることから、最良のパラメータ値ということになる。しかし、この ( $w, L$ ) で構成されたネットワークでは、記号表現ベクトルの区別ができず、ほぼすべての記号の同定に失敗

し、一般に考えられるどのようなタスクを設定したとしても正解率が低くなる。このことは、本論文 4.2 節で実験的にも示す。

### 2.3 提案モデル

上記のように、教師パターンが記号表現ベクトルに依存するネットワークにおいて、「平均二乗誤差」を小さくすることは正解率（入力記号列に対する望ましい出力記号列の同定に成功する割合）を高めることにつながらない。そこで本論文では、記号の同定まで考慮した以下の目的関数  $F$  に基づいた学習法を提案する。

$$F = \frac{1}{S_{out}} \sum_{s=0}^{S_{out}-1} \frac{1}{N} \sum_{p=1}^N F_p(s)$$

$S_{out}$  は出力スロット数、 $s$  はスロット番号である。 $F_p(s)$  は  $p$  番目の学習データに対する出力第  $s$  スロットの記号の同定ができれば小さな値、記号の同定に失敗するならば大きな値となるように設定する。 $F$  が最小（実際は極小）になるように  $w, L$  を学習する。

本論文では、 $F_p$  を基本的に以下のように設定している。

$$F_p(s) = \frac{1}{V-1} \sum_{c \neq tw_{p,s}} F_p(s, c)$$

$$F_p(s, c) = f_\alpha \left( \frac{E_p(s, tw_{p,s}) + \varepsilon}{E_p(s, c) + \varepsilon} - 1 \right) \quad (8)$$

$$E_p(s, c) = \frac{1}{m} \sum_{e=1}^m \frac{1}{2} \left( o_{p,sm+e}^{(3)} - r_{c,e} \right)^2$$

$$f_\alpha(z) = \frac{1}{1 + \exp(-\alpha z)}$$

前述のように、 $V$  は  $L$  中の記号の総数、 $m$  は記号表現ベクトルの次元数である。 $tw_{p,s}$  は、パターン  $p$  の出力第  $s$  スロットの教師記号のインデックス、つまり  $W_{tw_{p,s}}$  が教師記号である。 $\varepsilon$  は十分小さな正数である。

$E_p(s, c)$  は、出力第  $s$  スロットの出力ベクトルと記号  $W_c$  の記号表現ベクトルとの二乗距離である。 $f_\alpha(z)$  は、 $\alpha$  が十分大きいならばステップ関数に近づく。このとき  $F_p(s, c)$  は、 $p$  番目のデータに対する出力第  $s$  スロットのベクトルと教師第  $s$  スロットのベクトルに関して、

出力と教師の二乗距離  $<$

出力と記号  $c$  間の二乗距離

が成立すれば 0、成立しなければ 1 である。 $F_p(s)$  は上記が成立しない単語  $W_c$  ( $c \neq tw_{p,s}$ ) の数を  $W_c$

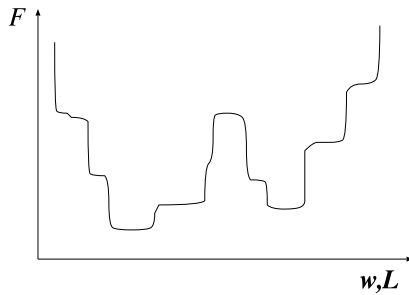


図2  $\alpha$  が大きいときの目的関数  $F$   
Fig.2 Objective function  $F$  with a large  $\alpha$ .

( $c \neq tw_{p,s}$ ) の総数  $V-1$  で割ったものであり,  $p$  番目のデータに対する第  $s$  スロットの出力が誤認識する記号の割合を意味する. したがって, 目的関数の値  $F$  は, 1 スロットあたりの誤認識する記号の割合を意味し,  $F$  が小さくなるようにパラメータを更新することで, 正解率 (入力記号列に対する望ましい出力記号列の同定に成功する割合) が高くなると考えられる.

$F$  の値が「1 スロットあたりの誤認識記号の割合」を意味するためには  $\alpha$  は十分大きくなければならない. ところが,  $\alpha$  が十分に大きいと, 目的関数はパラメータ空間のほとんどの領域で平坦になってしまい (図2), パラメータ学習が極端に遅くなる. そこで  $\alpha$  を学習ステップ  $\tau$  の関数として,

$$\alpha = \alpha_0 \tau \quad (10)$$

と設定し, 学習が進むにつれて大きな値となるようにしている.

パラメータ学習は最急降下法を用い, 各パラメータは,

$$w_{i,j}^{(l)} \leftarrow w_{i,j}^{(l)} - \eta \frac{\partial F}{\partial w_{i,j}^{(l)}},$$

$$L_d \leftarrow L_d - \eta \frac{\partial F}{\partial L_d} \quad (L_d \text{ は } L \text{ の第 } d \text{ 成分})$$

に従って更新される ( $\partial F / \partial w_{i,j}^{(l)}$ ,  $\partial F / \partial L_d$  の具体的な計算法については付録を参照).

学習の過程で, 偶然ある  $p, s, c$  ( $c \neq tw_{p,s}$ ) に

式 (8) を用いた目的関数とほぼ同様の効果は, ある  $\varepsilon (> 0)$  を設定して,

$$F_p(s, c) = f_\alpha(E_p(s, tw_{p,s}) - E_p(s, c) + \varepsilon) \quad (9)$$

としても得られるが, 式 (8) で定義された  $F_p(s, c)$  では, 第  $s$  スロットの出力ベクトルが第  $s$  スロットの教師信号のベクトルに十分近い場合,  $W_c$  ( $c \neq tw_{p,s}$ ) に対するベクトルは  $F_p(s, c)$  をほとんど大きくすることなく比較的自由に設定できるのに対し, 式 (9) で定義される  $F_p(s, c)$  では, 第  $s$  スロット出力ベクトルが第  $s$  スロット教師記号のベクトルに十分近い場合でも,  $W_c$  のベクトル設定に対する自由度は小さい. 予備的に行った実験では, 式 (9) の  $F_p(s, c)$  を用いた場合, 得られる記号表現ベクトルは均等に分布する傾向にあった.

対して  $E_p(s, c) = 0$  (あるいはきわめて小さな正数) となることが考えられる. この場合, 式 (8) 中に正定数  $\varepsilon$  が無い, つまり  $\varepsilon = 0$  とすると,  $\partial F / \partial w_{i,j}^{(l)}$ ,  $\partial F / \partial L_d$  の絶対値が  $\infty$  (あるいはきわめて大) となる (付録 A.1 参照). 式 (8) 中の正定数  $\varepsilon$  は  $E_p(s, c) = 0$  (あるいはきわめて小さな正数) となる場合でも, 学習が安定するように導入したものである.

### 3. 関連研究

本論文の提案モデルと密接な関係を持つモデルとして, Miikkulainen らの FGREP モジュール<sup>6)</sup> がある.

FGREP モジュールは, 自然言語処理研究分野における深層格解析等のタスクを学習しながら, 同時に記号表現ベクトル辞書を学習する. FGREP モジュールは多段に結合して用いることができ, より高度なタスクを実現する. この場合, 出力された記号表現ベクトルの記号の同定は行わず, 出力ベクトルが次のモジュールの入力ベクトルとなる.

深層格解析を行う FGREP モジュール (図3) の入力層には, 主語, 動詞, 目的語の名詞, with と結合する名詞の 4 つの入力スロットがあり, 各スロットに, 対応する記号の記号表現ベクトルを入力する. 図3は, “The girl ate pasta with a fork.” の文が入力された例であり, 各入力スロットには, ‘girl’, ‘ate’, ‘pasta’, ‘fork’ の単語の記号表現ベクトルが入力される. 出力層は, 動作主格 (agent), 動作 (act), 受動者格 (patient), 道具格 (inst), 修飾語 (mod) の 5 つの出力スロットからなり, 入力文に対して, 各スロットに, そのスロットが表す深層格を占める単語の記号表現ベクトル (に類似したベクトル) が出力されるように結合荷重および記号表現ベクトル辞書が学習される. 図3の例では, agent, act, patient, inst, mod スロットにはそれぞれ girl, ate, pasta, fork, NULL (なしを意味する特別な記号) の記号表現ベクトルに類似したベクトルが出力されることになる. 記号表現ベクトルの初期値はランダムに設定されたが, 深層格解析の学習が進行するに従って, 学習データ中の統語的, 意味的側面を反映した記号表現ベクトルが獲得されたと報告されている.

FGREP モジュールの学習は, ネットワーク構成と目的関数が MSE 基準モデルと同じため, 最急降下法に従うならば 2.2 節とまったく同じになるはずである. しかし, 文献 6) で示された更新式はそのようになっていない. 文献 6) で示された更新式は以下のとおりである. 結合荷重の更新式は通常の BP 法と同じである. 記号表現ベクトルの更新式は, 入力ユニットの入

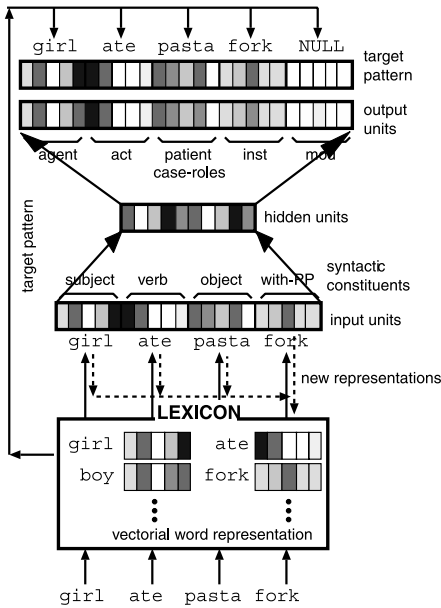


図3 FGREP モジュールの構成

Fig. 3 Structure of the FGREP module.

出力関数をシグモイド関数ではなく恒等関数と考えた上で、結合荷重の学習 (2), (3), (4) の類推から次のように与えられている (ただし、本論文と同じ表記を用いる)。

$$r_{c,e} \leftarrow r_{c,e} - \eta \frac{1}{N} \sum_{p=1}^N \sum_{i=1}^{n_1} \delta_{p,i}^{(1)} \cdot [M_p]_{i,cm+e} \quad (11)$$

$$\delta_{p,i}^{(1)} = \sum_{j=1}^{n_2} \delta_{p,j}^{(2)} \cdot w_{i,j}^{(2)} \quad (12)$$

ところが、この更新式は式 (1) を目的関数とする最急降下法に基づいて得られる更新式ではない。式 (11) と式 (5), (6) を比較すると、式 (11) には式 (6) の第 2 項に相当する項がない。FGREP モジュールにおける学習は 2.2 節で述べた問題を避けるため、あえて式 (6) の第 2 項を除いているとも考えられる。つまり、辞書を便宜上、入力ベクトルを合成するための辞書  $L_{in}$  と、教師ベクトルを合成するための辞書  $L_t$  とに分け、第  $k$  ステップでの記号表現ベクトルの学習 (更新) では、 $L_t$  を第  $k-1$  ステップで得られた  $L_{in}$  に固定し、 $L_{in}$  を学習した後、 $L_t \leftarrow L_{in}$  とする。2.2 節で述べたような無意味な解が得られる可能性があることは、Miikkulainen も指摘しているが、確かに追実験では、彼らの報告のように、そのような解には陥らなかった。しかし、数学的な意味が不明である。

さらに、出力側での記号の同定の正解率は、4.4 節で示すようにきわめて低い。文献 (6) では学習された  $w$  や  $L$  の評価の 1 つとして、各ユニットの出力値と教師信号の値との誤差が 0.15 未満である割合を取り上げ、高い値であったと報告している。彼らは、FGREP モジュールを多段に結合して物語パラフレイザ DISPAR を構築しており、下位の FGREP モジュールの出力ベクトルがそのまま上位の FGREP モジュールの入力ベクトルとなるため、必ずしも本論文で用いた出力側での記号の同定の正解率で評価する必要はない。しかし、たとえば man, woman, boy, girl の 4 つの単語に対する獲得された記号表現ベクトルは互いにきわめて類似しており、上位の FGREP モジュールの性能低下をまねく。これを防ぐため、Miikkulainen らは、2 ビットの識別ビットを記号表現ベクトルに付加する “ID-cloning” (学習中も変化しない、異なる語には異なる識別ビットを付加) と呼ぶ手法を用いているが、きわめて ad hoc である。

#### 4. 実験

本章では、MSE 基準モデル、記号表現ベクトルを学習しないモデル (SBP モデル)、提案モデル、FGREP モジュールに関する実験について説明する。

##### 4.1 データ、評価法および実験条件

使用した記号列データは、文献 (6) で用いられたデータとほぼ同様の人工的な言語データであり、タスクは深層格解析である (名詞、動詞の異なり数はそれぞれ 24 語、4 語)。データは文テンプレート (表 1) とメタワード (表 2) から作成した。たとえば、テンプレート 4 とメタワードの表から、表 3 に示す  $4 \times 4 \times 2$  個のデータが作成される (入力は、主語、動詞、目的語の名詞、with と結合する名詞の順、出力は、動作主格、動作、受動者格、道具格、修飾語の順)。文献 (6) と同様に、すべての組合せより全部で 1,275 文を作成した。MSE 基準モデルの問題を確認する実験、FGREP モジュールの動作を確認する実験では、このうち 90% (1,147 文) を学習データとして、残りの 10% (128 文) をテストデータとして用いた。可能なすべての入力パターンのうち 90% が学習データとして与えられるというのは現実的ではない。そこで、提案モデルの有効性を示すため、提案モデルに関する実験および比較実験である SBP モデルに関する実験では、全 1,275 文のうち 10% (128 文) を学習データとして、残りの 90% (1,147 文) をテストデータとして用いた実験も行った。モデルの性能を測る指標には、文正解率を用いる。文正解率とは、1 文 5 スロットすべて記号の同

表 1 文テンプレート  
Table 1 Sentence templates used in the experiments.

No.	template				correct case-role		
	subject	verb	object	with-PP	subject	object	with-PP
1	HUMAN	ate			agent		
2	HUMAN	ate	FOOD		agent	patient	
3	HUMAN	ate	FOOD	FOOD	agent	patient	modifier
4	HUMAN	ate	FOOD	UTENSIL	agent	patient	instrument
5	ANIMAL	ate			agent		
6	PREDATOR	ate	PREY		agent	patient	
7	HUMAN	broke	FRAGILE		agent	patient	
8	HUMAN	broke	FRAGILE	BREAKER	agent	patient	instrument
9	BREAKER	broke	FRAGILE		instrument	patient	
10	ANIMAL	broke	FRAGILE		agent	patient	
11	FRAGILE	broke			patient		
12	HUMAN	hit	THING		agent	patient	
13	HUMAN	hit	HUMAN	POSSESSION	agent	patient	modifier
14	HUMAN	hit	THING2	HITTER	agent	patient	instrument
15	HITTER	hit	THING		instrument	patient	
16	HUMAN	moved			agt+pat		
17	HUMAN	moved	OBJECT		agent	patient	
18	ANIMAL	moved			agt+pat		
19	OBJECT	moved			patient		

表 2 メタワード

Table 2 Correspondence between meta-words and actual words.

meta-word	nouns
HUMAN	man woman boy girl
ANIMAL	bat chicken dog sheep wolf lion
PREDATOR	wolf lion
PREY	chicken sheep
FOOD	chicken cheese pasta carrot
UTENSIL	fork spoon
FRAGILE	plate window vase
HITTER	bat ball hatchet hammer vase paperwt rock
BREAKER	bat ball hatchet hammer paperwt rock
POSSESSION	bat ball hatchet hammer vase dog doll
OBJECT	bat ball hatchet hammer paperwt rock vase plate window fork spoon pasta cheese chicken carrot doll
THING	HUMAN ANIMAL OBJECT
THING2	ANIMAL OBJECT

表 3 テンプレート 4 から作成されるデータ例

Table 3 Example of data generated from sentence template 4.

input	output
man ate chicken fork	man ate chicken fork NULL
man ate chicken spoon	man ate chicken spoon NULL
.	.
.	.
girl ate carrot spoon	girl ate carrot spoon NULL

定に成功すれば正解とした正解率である。

名詞 24 語はメタワードが同一の単語が 7 グループあり、それらは  $\{man, woman, boy, girl\}$ ,  $\{wolf, lion\}$ ,  $\{cheese, carrot, pasta\}$ ,  $\{window, plate\}$ ,  $\{fork,$

$spoon\}$ ,  $\{hatchet, ball, hammer\}$ ,  $\{rock, paperwt (paperweight, 文鎮)\}$  である。これら 7 グループの記号表現ベクトルは記号表現ベクトルの次元数  $m$ , 中間ユニット数  $n_2$  が適切であるならば、学習により類似したベクトルになるべきである。これをモデルの性能を測るもう 1 つの指標とした。

実験条件は、すべてのモデルで、中間ユニット数  $n_2$  は 12, 記号表現ベクトルの次元数  $m$  は 3, 結合荷重の初期値は  $[-1, 1]$  の一様乱数とした。MSE 基準モデル, 提案モデル, FGREP モジュールに関する実験では、記号表現ベクトルの各要素は、 $[0, 1]$  の一様乱数に初期化し、学習中は  $[0, 1]$  の範囲に収まる制限をかけた。この学習をパラメータ設定のための一様乱数を変えて 5 種類試し、目的関数の値が最小となる学習結果を最終結果とした。以上の実験条件に加えて、各モデル固有の実験条件についてのみ、以下の節で示すことにする。

#### 4.2 MSE 基準モデル

2.2 節で示した MSE 基準モデルの問題を確認するための実験を行った。

学習係数  $\eta$  は 0.05 に固定した。

実験結果として、図 4 に目的関数の値  $E$  と文正解率の推移を示す。学習が始まるとすぐに  $E$  は 0 に収束し始める。学習が 4,400 回の時点で、24 個の名詞の記号表現ベクトルはすべて  $[0.393, 0.409, 0.472]$  になり、4 個の動詞の記号表現ベクトルはすべて  $[0.697, 0.386, 0.604]$  になった。この  $L$  は、 $\forall p M_p L = x_{in}$ ,  $\forall p T_p L = x_t$  となっていることが分かる。また、

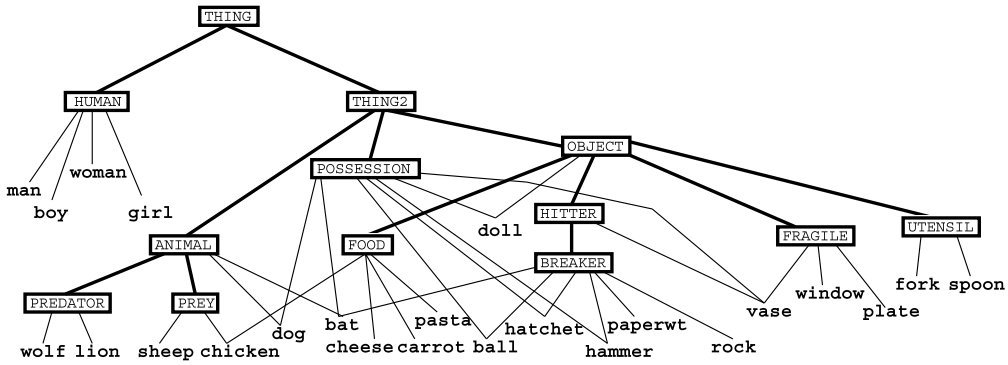


図 5 メタワード (表 2) から求められるシソーラス  
 Fig. 5 Thesaurus expressed by the meta-words (Table 2).

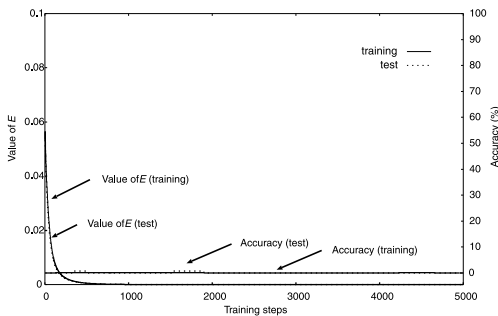


図 4 MSE 基準モデルの学習過程  
 Fig. 4 Learning process of the MSE model.

文正解率はきわめて低く、ほぼ 0%であった。パラメータの初期乱数を 5 種類試しても同様な結果となった。

この結果より、2.2 節で述べたように、目的関数の値  $E$  を最小にするような、タスクに対して無意味な記号表現ベクトルが学習され、記号の同定率が非常に低くなることが確認された。

4.3 記号表現ベクトルを学習しないモデル (SBP)

次に、ネットワーク構成はそのまま記号表現ベクトルを学習しない (学習パラメータが  $w$  のみの) 通常の BP 法について実験を行った。

学習係数  $\eta$  は 0.05 に固定し、100 万回学習を行った。記号表現ベクトルは 3 種類試した。1 つは  $[0, 1]$  の一様乱数に基づく  $L_r$  である。残りの 2 つは、動詞の記号表現ベクトルは  $[0, 1]$  上の一様乱数に従って設定し、名詞の記号表現ベクトルは、メタワード (表 2) 間の階層関係を反映するように設定した  $L_1$  と  $L_2$  であり、以下のようにして作成した。

(1) 表 2 で示されるメタワード間の階層関係を求め、図 5 で示すシソーラスを作成した。メタワード間の階層関係は、表 2 から各メタワード  $M_i$  のインスタンスの単語の集合  $Words(M_i)$  を求め、

$Words(M_i) \subset Words(M_j) \implies$   
 $M_i$  の上位のメタワードは  $M_j$  である  
 として設定した。

(2) 上記のシソーラスを基に、名詞間の類似度を求めた。1 つの名詞が複数の直接の上位メタワードを持つ場合があるため、類似度は、

$$\text{sim}(W_1, W_2) = \frac{1}{|U(W_1)| \times |U(W_2)|} \cdot \sum_{\substack{M_1 \in U(W_1) \\ M_2 \in U(W_2)}} \text{SIM}(W_1 : M_1, W_2 : M_2) \quad (13)$$

と定義した。ただし、 $U(W)$  は名詞  $W$  の直接の上位メタワードの集合であり、 $\text{SIM}(W_1 : M_1, W_2 : M_2)$  は、 $W_1, W_2$  の直接の上位メタワードをそれぞれ  $M_1, M_2$  とした場合の、 $W_1$  と  $W_2$  の類似度である。つまり、 $\text{sim}(W_1, W_2)$  は  $W_1, W_2$  の直接の上位メタワード  $M_1, M_2$  をすべて考えた場合の  $\text{SIM}(W_1 : M_1, W_2 : M_2)$  の平均になっている。今回は、 $\text{SIM}(W_1 : M_1, W_2 : M_2)$  として以下の 2 つの類似度  $\text{SIM}_1$  と  $\text{SIM}_2$  を試みた。 $\text{SIM}_1$  は、基本的には文献 11) で紹介されている類似度で、

$$\text{SIM}_1(W_1 : M_1, W_2 : M_2) = \frac{2d_c(M_1, M_2)}{d(W_1 : M_1) + d(W_2 : M_2)} \quad (14)$$

である。ここで、 $d_c(M_1, M_2)$  は  $M_1, M_2$  の共通の上位ノードの深さ (ルートノードからのパス長)、 $d(W : M)$  は直接の上位メタワードを  $M$  とした場合の単語  $W$  の深さであり、 $M$  の深さ +1 である。 $\text{SIM}_2$  は、

$$\text{SIM}_2(W_1 : M_1, W_2 : M_2) = -L(W_1 : M_1, W_2 : M_2)$$

である。ここで、 $L(W_1 : M_1, W_2 : M_2)$  は  $W_1, W_2$



の直接の上位メタワードをそれぞれ  $M_1, M_2$  とした場合の  $W_1$  と  $W_2$  間のパス長であり,  $M_1$  と  $M_2$  間のパス長 +2 である. SIM<sub>1</sub> を用いて後述する手順 (3), 手順 (4) を行って作成した記号表現ベクトル辞書が  $L_1$  であり, SIM<sub>2</sub> を用いた場合の記号表現ベクトル辞書が  $L_2$  である.

(3) (2) で求めた名詞間の類似度から数量化 IV 類を用いて, 各名詞  $W_c$  に対して暫定的な 3 次元の記号表現ベクトル  $r'_c$  を求めた. 数量化 IV 類は, 個体間の類似度を基に各個体をユークリッド空間上に配置する方法の 1 つで, 個体間の類似度が高い個体どうしのベクトルは互いに近くに, 個体間の類似度が低い個体どうしのベクトルは互いに遠くに配置される<sup>12)</sup>.

(4) (3) で求めたベクトルは同一のベクトルになる名詞がある. ベクトルが完全に一致すれば, 当然記号の同定の正解率は低くなる. そこで, 以下のようにして (3) で求めたベクトルにランダムノイズを付加する.

(a) 各  $e (e \in \{1, 2, 3\})$  に対して, 名詞の暫定的記号表現ベクトル間の第  $e$  成分の最小差 (0 は除く)  $MD_e$

$$MD_e = \min_{\substack{c_1, c_2 \\ (r'_{c_1, e} \neq r'_{c_2, e})}} |r'_{c_1, e} - r'_{c_2, e}|$$

を求める.

(b) 各名詞  $W_{c_1}$ , 各  $e (e \in \{1, 2, 3\})$  に対して,  $r'_{c_1, e} = r'_{c_2, e}$  なる  $W_{c_2}$  が存在すれば,

$$r''_{c_1, e} = r'_{c_1, e} + \frac{MD_e}{2} \times Ra,$$

存在しないならば,

$$r''_{c_1, e} = r'_{c_1, e}$$

とする. ただし,  $Ra$  は  $[-1, 1]$  上の一様乱数である.

(c) 各名詞  $W_c$ , 各  $e (e \in \{1, 2, 3\})$  に対して,

$$r_{c, e} = \frac{r''_{c, e} - \min_{W_c} r''_{c, e}}{\max_{W_c} r''_{c, e} - \min_{W_c} r''_{c, e}} \quad (15)$$

のように  $r''_{c, e}$  を  $[0, 1]$  区間に線形変換して記号表現ベクトル  $r_c$  を得る. 動詞に対しては  $[0, 1]$  上の一様乱数に従って記号表現ベクトルを設定する. 結果を表 4 (SBP の行) に示す. 当初, 記号表現ベクトルを学習しないモデルで, 記号表現ベクトル辞書としてメタワード間の階層関係を反映した  $L_1, L_2$  を

用いた場合は, 一様乱数により生成した  $L_r$  を用いた場合より正解率が高くなると予想したが, そのような結果は得られなかった. これは,  $L_1, L_2$  の求め方に問題があるわけだが,  $L_1, L_2$  は, 尤もらしい記号表現ベクトル辞書の候補であることは確かである. この結果だけから, メタワードの階層関係のような別の知識を基に  $L$  を求める方法が良くないとは断言できないが, 後述する提案モデルの結果程度に良い結果が得られるような記号表現ベクトルを求めるのはかなり難しいと考えられる.

4.4 提案モデル

次に, 提案モデルの性質を調べるための実験を行った.

式 (8) の  $\epsilon$  は  $10^{-20}$  とし, 式 (10) の比例定数  $\alpha_0$  は  $1/20000$  とした.  $\partial F / \partial w_{i,j}^{(l)}, \partial F / \partial L_d$  は付録に示すように,  $\alpha$  に比例する. 学習ステップ  $\tau$  が大きくなったとき, 更新量が大きくなって学習が不安定になることを防ぐために, 学習率  $\eta$  は,

$$\eta = \eta_0 / \alpha$$

として学習中に変化させた. 本実験では  $\eta_0 = 0.2$  とした. また, 学習は予備実験により, 十分  $F$  が収束したと考えられた 30 万回まで行った.

学習データ数: テストデータ数が 9 : 1 の場合の学習データに対する学習過程 (目的関数の値  $F$  の推移と文正解率の推移) を図 6 に示す. また, テストデータに対する学習過程を図 7 に示す. 目的関数の値  $F$

表 4 テストデータに対する文正解率 (%)  
Table 4 Sentence accuracy for test data (%).

学習法	学習データ対テストデータ比	
	9 : 1	1 : 9
SBP( $L_r$ )	73.2	49.7
SBP( $L_1$ )	3.15	3.40
SBP( $L_2$ )	33.9	33.2
提案モデル	96.1	72.3

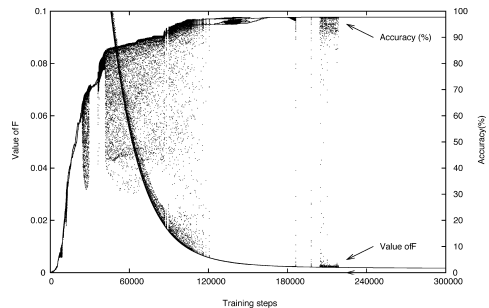


図 6 提案モデルの学習過程 (学習データ)

Fig. 6 Learning process of the proposed model (training data).

記号表現ベクトル辞書が  $L_r$  に関しては,  $n_2 = 24$  とした場合についても実験を行ったが,  $n_2 = 12$  とした場合よりもテストデータに対する文正解率は低くなった.

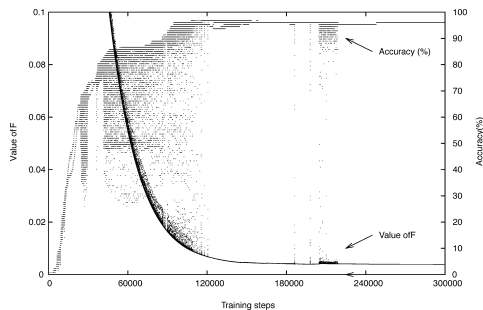


図 7 提案モデルの学習過程 (テストデータ)

Fig. 7 Learning process of the proposed model (test data).

は学習ステップが約 20 万回程度まで不安定に値が上下しているが、約 20 万回以降は安定して減少している。文正解率の推移は  $F$  の減少にもなって増加している。約 20 万回までは正解率も不安定であるが、その後は安定しており、学習 30 万回で学習データに対して 97.7%、テストデータに対して 96.1% の結果となった。

以上の結果から、提案モデルは学習初期には不安定な挙動を示すが、学習が進むと高い文正解率を示すことが分かる。学習初期の不安定な挙動を示す問題は、学習の更新量が大きすぎることが原因であると考えられるので、学習係数  $\eta$ 、すなわち  $\eta_0$  を小さくすれば解消されると考えてよい。

学習データ数：テストデータ数が 9:1 および 1:9 の場合のテストデータに対する文正解率を表 4 (提案モデルの行) に示す。表 4 より、提案モデルはテストデータに対し高い文正解率を示す、つまり高い予測能力を有することが分かる。

次に、学習 30 万回時の記号表現ベクトルについて、クラスタ分析 (群平均法) を行った結果を図 8 に示す。名詞と動詞は分かれており、さらに 4.1 節で説明したメタワードが同一の名詞 7 グループは、ほぼ近隣に位置していることが分かる。このことは、類似の使われかたをする 2 つの単語の記号表現ベクトルどうしが近いことを示している。このことに加えて、学習データとテストデータの性質が似ているため、提案モデルは、テストデータに対しても高い文正解率を示したと考えられる。

#### 4.5 FGREP モジュール

次に、3 章で説明した FGREP モジュールの特徴を確認するための実験を行った。

作成データすべてを学習データとしているわけではないので、メタワードが同一の名詞 7 グループはそれぞれ、学習データ中においてまったく同じ使われ方をするわけではない。

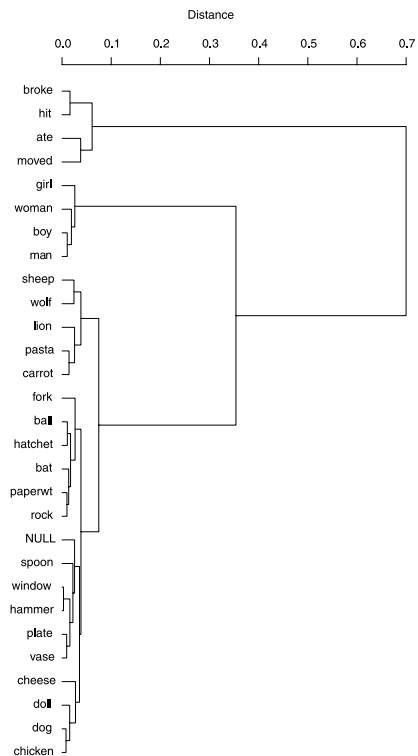


図 8 クラスタ分析結果 (樹状図)

Fig. 8 Result of cluster analysis (dendrogram).

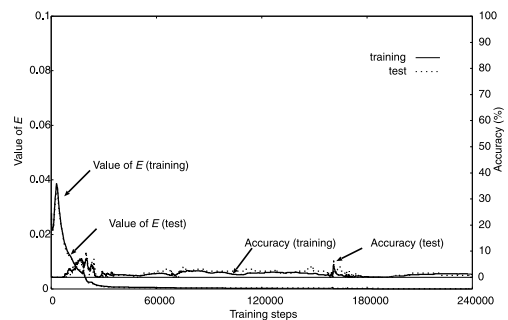


図 9 FGREP モジュールの学習過程

Fig. 9 Learning process of the FGREP module.

学習係数  $\eta$  は 0.05 に固定し、予備実験により十分に学習が経過したと判断した 24 万回まで学習を行った。

まず、他のモデルの実験条件と同様の条件とした実験の結果を図 9 に示す。学習が進むに従い、目的関数の値  $E$  は全体として減少してはいるがその変化は不安定である。さらに、単語の同定を行ったときの文正解率は非常に低い。これらのことは、3 章で説明したように、学習が真の最急降法に従っていないこと、単語の同定を考慮した目的関数ではないことに起因する。

次に、Miikkulainen らの原著文献<sup>6)</sup> の追試について述べる。中間ユニット数  $n_2$  は 25 とし、記号表現ベ

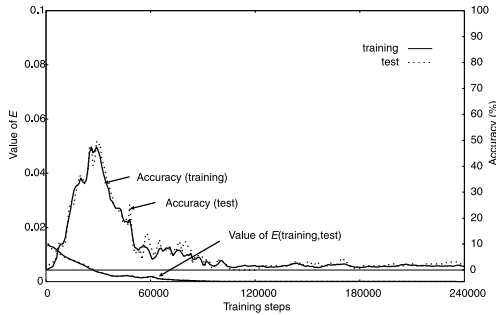


図 10 FGREP モジュール (追試) の学習過程

Fig. 10 Learning process of the original FGREP module.

クトルの次元数  $m$  は 12 とした。結合荷重の初期値は  $[-1, 1]$  の一様乱数とした。記号表現ベクトルの各要素は、 $[0, 1]$  の一様乱数に初期化し、学習中は  $[0, 1]$  の範囲に収まる制限をかけた。学習係数  $\eta$  は、学習開始の 200 回は 0.01, 200 回以降 500 回までは 0.05, 500 回以降は 0.025 とし、逐次学習を行った。他の実験と同様、学習パラメータの初期値を 5 種類試し、最も小さい  $E$  を示したときの学習過程を図 10 に示す。図 9 と同様に、目的関数の値の変化は非常に不安定であり、単語の同定を行ったときの文正解率は非常に低い。

## 5. おわりに

本論文では、記号列を入力とする関数をニューラルネットワークを用いて学習する手法について述べた。提案モデルは、記号表現ベクトルも学習することが特徴で、出力側での記号の同定を考慮した目的関数を用いることで、学習データの性質を反映した記号表現ベクトルを獲得する。ただし、今回検証実験に用いた深層格解析のタスクでは、出力記号列の記号列はすべて入力記号列に現れる記号であり、入力記号列と出力記号列が別の記号集合からなるタスクの性能を検証したわけではない。今後、入力記号列と出力記号列が別の記号集合からなるデータに対する検証が課題である。

## 参考文献

- 1) 中川聖一: 情報科学コアカリキュラム講座, 丸善 (1999).
- 2) Schmid, H.: Part-of-speech tagging with neural networks, *Proc. COLING-94*, pp.172-176 (1994).
- 3) Sopena, J., Lloberas, A. and Moliner, J.: A Connectionist Approach to Prepositional Phrase Attachment for Real World Texts, *Proc. COLING-ACL '98*, pp.1233-1237 (1998).
- 4) Alegre, M., Lloberas, A. and Sopena, J.: Syn-

tactic Disambiguation: A Neural Network Approach, *Proc. ATALA, Workshop Representation and Treatment of Syntactic Ambiguity in Natural Language Processing*, Paris, pp.11-17 (2000).

- 5) Dale, R., Moisl, H. and Somers, H.: *Artificial Neural Network Approaches to NLP*, Marcel Dekker Inc. (2000).
- 6) Miikkulainen, R. and Dyer, M.G.: Natural Language Processing With Modular PDP Networks and Distributed Lexicon, *Cognitive Science*, Vol.15, pp.343-399 (1991).
- 7) Takahashi, N. and Motoki, M.: A subsymbolic approach for acquiring semantic representations, *3rd International Workshop on Computational Semantics* (1999).
- 8) Tomiura, Y., Tanaka, S. and Hitaka, T.: Estimating satisfactoriness of selectional restriction from corpus without a thesaurus, *ACM Trans. Asian Language Information Processing*, Vol.4, pp.400-416 (2005).
- 9) Kleiweg, P. and Nerbonne, J.: An FGREP investigation into phonotactics, *CLIN'98, Language and Computers: Studies in Practical Linguistics*, Vol.29, pp.37-50 (1999).
- 10) Rumelhart, D., Hinton, G. and Williams, R.: *Learning Internal Representations by Error Propagation*, Vol.1, The MIT Press, Massachusetts (1986).
- 11) 長尾 真 (編): 岩波ソフトウェア科学講座 15 自然言語処理, 岩波書店 (1996).
- 12) 林知己夫: 数量化—理論と方法, 朝倉書店 (1993).

## 付 録

### A.1 提案モデルの学習の詳細

各パラメータの学習は最急降下法により行われる。 $E_p(s, c)$  の各パラメータに関する偏導関数は以下のように求まる。

BP 法のように誤差信号  $\delta$  を導入し,

$$\begin{aligned} \delta_p^{(3)}(s, e, c) &\stackrel{\text{def}}{=} \frac{\partial E_p(s, c)}{\partial u_{p, sm+e}^{(3)}} \\ &= \frac{\partial E_p(s, c)}{\partial o_{p, sm+e}^{(3)}} \cdot \frac{\partial o_{p, sm+e}^{(3)}}{\partial u_{p, sm+e}^{(3)}} \\ &= \frac{1}{m} \cdot (o_{p, sm+e}^{(3)} - r_{c,e}) o_{p, sm+e}^{(3)} \cdot (1 - o_{p, sm+e}^{(3)}) \end{aligned}$$

とおくと,

$$\begin{aligned} \frac{\partial E_p(s, c)}{\partial w_{j, sm+e}^{(3)}} &= \frac{\partial E_p(s, c)}{\partial u_{p, sm+e}^{(3)}} \cdot \frac{\partial u_{p, sm+e}^{(3)}}{\partial w_{j, sm+e}^{(3)}} \\ &= \delta_p^{(3)}(s, e, c) \cdot o_{p,j}^{(2)} \end{aligned}$$

$$\left( \begin{array}{c} \frac{\partial u_{p,sm+e}^{(3)}}{\partial w_{j,sm+e}^{(3)}} = o_{p,j}^{(2)} \end{array} \right). \quad (16)$$

$$\begin{aligned} \frac{\partial E_p(s, c)}{\partial b_{sm+e}^{(3)}} &= \frac{\partial E_p(s, c)}{\partial u_{p,sm+e}^{(3)}} \cdot \frac{\partial u_{p,sm+e}^{(3)}}{\partial b_{sm+e}^{(3)}} \\ &= \delta_p^{(3)}(s, e, c) \\ &\left( \begin{array}{c} \frac{\partial u_{p,sm+e}^{(3)}}{\partial b_{sm+e}^{(3)}} = 1 \end{array} \right). \end{aligned} \quad (17)$$

また,

$$\begin{aligned} \delta_p^{(2)}(s, j, c) &\stackrel{\text{def}}{=} \frac{\partial E_p(s, c)}{\partial u_{p,j}^{(2)}} \\ &= o_{p,j}^{(2)} \cdot (1 - o_{p,j}^{(2)}) \cdot \sum_{e=1}^m \delta_p^{(3)}(s, e, c) \cdot w_{j,sm+e}^{(3)} \end{aligned}$$

とあくと,

$$\begin{aligned} \frac{\partial E_p(s, c)}{\partial w_{i,j}^{(2)}} &= \frac{\partial E_p(s, c)}{\partial u_{p,j}^{(2)}} \cdot \frac{\partial u_{p,j}^{(2)}}{\partial w_{i,j}^{(2)}} \\ &= \delta_p^{(2)}(s, j, c) \cdot o_{p,i}^{(1)}. \end{aligned} \quad (18)$$

$$\begin{aligned} \frac{\partial E_p(s, c)}{\partial b_j^{(2)}} &= \frac{\partial E_p(s, c)}{\partial u_{p,j}^{(2)}} \cdot \frac{\partial u_{p,j}^{(2)}}{\partial b_j^{(2)}} \\ &= \delta_p^{(2)}(s, j, c). \end{aligned} \quad (19)$$

また,

$$\begin{aligned} \delta_p^{(1)}(s, i, c) &\stackrel{\text{def}}{=} \frac{\partial E_p(s, c)}{\partial u_{p,i}^{(1)}} \\ &= \sum_{e=1}^m \delta_p^{(2)}(s, j, c) \cdot w_{i,j}^{(2)} \end{aligned}$$

とおき, 記号表現ベクトル辞書  $L$  の第  $d$  成分を  $L_d$  で表すと,

$$\begin{aligned} \frac{\partial E_p(s, c)}{\partial L_d} &= \sum_i \frac{\partial E_p(s, c)}{\partial u_{p,i}^{(1)}} \cdot \frac{\partial u_{p,i}^{(1)}}{\partial L_d} + \sum_{e=1}^m \frac{\partial E_p(s, c)}{\partial r_{c,e}} \frac{\partial r_{c,e}}{\partial L_d} \\ &= \sum_{i=1}^{n_1} \delta_p^{(1)}(s, i, c) \cdot [M_p]_{i,d} \\ &\quad - \frac{1}{m} \sum_{e=1}^m I(d = cm + e) (o_{p,sm+e}^{(3)} - r_{c,e}). \end{aligned} \quad (20)$$

ただし,  $I(P)$  は命題  $P$  が真のときのみ 1, ほかは 0 となる関数である.

$F_p(s, c)$  に対するパラメータ  $y$  ( $y$  は,  $w_{i,j}^{(l)}$ ,  $b_j^{(l)}$  または  $L_d$ ) の偏導関数は,

$$\begin{aligned} \frac{\partial F}{\partial y} &= \frac{1}{S_{out} \cdot N \cdot (V - 1)} \\ &\quad \cdot \sum_{s=0}^{S_{out}-1} \sum_{p=1}^N \sum_{c \neq tw_{p,s}} \frac{\partial F_p(s, c)}{\partial y} \end{aligned} \quad (21)$$

であり, また  $F_p(s, c)$  に対するパラメータ  $y$  の偏導関数は,

$$\begin{aligned} \frac{\partial F_p(s, c)}{\partial y} &= \alpha \cdot F_p(s, c) \cdot (1 - F_p(s, c)) \\ &\quad \cdot \left\{ \frac{1}{E_p(s, c) + \varepsilon} \cdot \frac{\partial E_p(s, tw_{p,s})}{\partial y} \right. \\ &\quad \left. - \frac{E_p(s, tw_{p,s}) + \varepsilon}{(E_p(s, c) + \varepsilon)^2} \cdot \frac{\partial E_p(s, c)}{\partial y} \right\} \end{aligned} \quad (22)$$

$$\begin{aligned} \left( \because f'_\alpha(z) = \frac{\alpha \cdot e^{-\alpha z}}{(1 + e^{-\alpha z})^2} \right. \\ \left. = \alpha \cdot f_\alpha(z) \cdot (1 - f_\alpha(z)) \right) \end{aligned}$$

である. したがって式 (16), (21), (22) より,

$$\begin{aligned} \frac{\partial F}{\partial w_{j,sm+e}^{(3)}} &= \frac{\alpha}{S_{out} \cdot N \cdot (V - 1)} \\ &\quad \cdot \sum_{s=0}^{S_{out}-1} \sum_{p=1}^N \sum_{c \neq tw_{p,s}} F_p(s, c) \cdot (1 - F_p(s, c)) \cdot o_{p,j}^{(2)} \\ &\quad \cdot \left\{ \frac{1}{E_p(s, c) + \varepsilon} \cdot \delta_p^{(3)}(s, e, tw_{p,s}) \right. \\ &\quad \left. - \frac{E_p(s, tw_{p,s}) + \varepsilon}{(E_p(s, c) + \varepsilon)^2} \cdot \delta_p^{(3)}(s, e, c) \right\}. \end{aligned}$$

同様に, 式 (17) ~ (22) より

$$\frac{\partial F}{\partial b_{sm+e}^{(3)}}, \frac{\partial F}{\partial w_{i,j}^{(2)}}, \frac{\partial F}{\partial b_j^{(2)}}, \frac{\partial F}{\partial L_d}$$

が求まる. これらを用いて以下の更新式によりパラメータ学習を行う.

$$w_{j,sm+e}^{(3)} \leftarrow w_{j,sm+e}^{(3)} - \eta \frac{\partial F}{\partial w_{j,sm+e}^{(3)}},$$

$$b_{sm+e}^{(3)} \leftarrow b_{sm+e}^{(3)} - \eta \frac{\partial F}{\partial b_{sm+e}^{(3)}},$$

$$w_{i,j}^{(2)} \leftarrow w_{i,j}^{(2)} - \eta \frac{\partial F}{\partial w_{i,j}^{(2)}},$$

$$b_j^{(2)} \leftarrow b_j^{(2)} - \eta \frac{\partial F}{\partial b_j^{(2)}},$$

$$L_d \leftarrow L_d - \eta \frac{\partial F}{\partial L_d}.$$

(平成 17 年 9 月 26 日受付)

(平成 18 年 5 月 9 日採録)



本木 実 (正会員)

1995年熊本大学大学院工学研究科電気情報工学専攻修士課程修了。1999年より九州産業大学工学部電気工学科助手。2003年10月九州大学大学院システム情報科学府博士課程入学，現在に至る。電子情報通信学会，言語処理学会各会員。

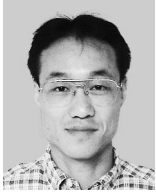
電子情報通信学会，言語処理学会各会員。



高橋 直人 (正会員)

筑波大学大学院博士課程工学研究科修了。博士(工学)。独立行政法人産業技術総合研究所情報技術研究部門主任研究員。多言語情報処理システム，自然言語処理等の研究に従事。言語処理学会，人工知能学会各会員。

言語処理学会，人工知能学会各会員。



富浦 洋一 (正会員)

1989年九州大学大学院工学研究科電子工学専攻博士課程単位取得退学。同年九州大学工学部助手，1995年同助教授，1996年同大学大学院システム情報科学研究科助教授，2000

年同大学院システム情報科学研究院助教授，現在に至る。博士(工学)。自然言語処理，計算言語学，人工知能に関する研究に従事。言語処理学会，人工知能学会各会員。

