記号表現ベクトル学習機能を有するニューラルネットの英語前置詞句係り先決定問題における実験的評価

本木, 実
九州大学大学院システム情報科学府知能システム学専攻 : 博士後期課程

冨浦, 洋一
九州大学大学院システム情報科学研究院知能システム学部門

日高, 達
九州大学名誉教授

嶋津, 好生
九州産業大学工学部電気工学科

他

https://doi.org/10.15017/1516061

出版情報：九州大学大学院システム情報科学紀要. 10 (1), pp.51-56, 2005-03-25. 九州大学大学院システム情報科学研究院
バージョン：
権利関係：

九州大学大学院
システム情報科学紀要
第10巻 第1号 平成17年3月

Research Reports on Information Science and
Electrical Engineering of Kyushu University
Vol.10, No.1, March 2005

記号表現ベクトル学習機能を有するニューラルネットの英語前置詞句係り先決定問題における実験的評価

本木 実* ・ 冨浦 洋一** ・ 日高 達***
嶋津 好生† ・ 高橋 直人††

An Evaluation of a Layered Neural Network which Have Function of Learning Vectorial Symbol Representations on PP-attachment Ambiguity Resolution

Minoru MOTOKI, Yoichi TOMIURA, Toru HITAKA,
Yoshio SHIMAZU and Naoto TAKAHASHI

(Received December 24, 2004)

Abstract: This paper describes a PP-attachment ambiguity resolution with a layered neural network which have function of learning vectorial symbol representations. The proposed model does not update only link weight but also vectorial symbol representations. We show qualitative difference between a proposed model and an ordinary layered neural network, which has more hidden units (i.e. more parameters) to have more flexibility but does not update symbol representations.

Keywords: PP-attachment, Neural network, Vectorial symbol representation

1. はじめに

英語前置詞句係り先決定問題（以下，PP-attachment問題）は，構文解析の一部に位置づけられ，自然言語処理において重要な問題の一つである．英語における以下の形態の動詞句

他動詞 + 名詞句1 + 前置詞 + 名詞句2

では，前置詞句（「前置詞 + 名詞句2」）の統語的な係り先として，他動詞と名詞句1の両方が可能である．例えば，今，次のような2文があるとする．

1. The girl ate pasta with a fork.
2. The girl ate pasta with cheese.

2文とも前置詞句の中の名詞句以外は全く同じ形態の文であり，前置詞句が動詞'ate'に係るか，目的語の名詞'pasta'に係るかの曖昧性が存在する．正しくは，文1では前置詞句 "with a fork"は動詞 'ate'に係り，文2では前置詞句 "with cheese"は名詞 'pasta'に係る．

PP-attachment問題を解く場合は，構文的な知識だけでなく意味的な知識が必要となる．本研究では，単語に対応する実ベクトル（以下，記号表現ベクトル)を単語の意味表現と考え，パターン情報を統計的に取り扱うことに優れた階層型ニューラルネットを用いる．ただし，通常の階層型ニューラルネットではなく記号表現ベクトル学習型ニューラルネット[1)]を用いる．本手法の学習は，通常の階層型ニューラルネットで用いられている通常のBP法を拡張し，結合荷重だけでなく記号表現ベクトルも学習する．本稿では，本手法の学習方法，および，PP-attachment問題に対する本手法と通常のBP法との比較実験について報告する．

平成16年12月24日受付
* 知能システム学専攻博士後期課程
** 知能システム学部門
*** 九州大学名誉教授
† 九州産業大学工学部電気工学科
†† 独立行政法人産業技術総合研究所(AIST)

2. 記号表現ベクトル学習型ニューラルネット

2.1 記号列を入力する階層型ニューラルネット

本稿で取り扱う階層型ニューラルネットが受け取る入力パターンは，入力記号列をその記号に対応するベクトルで置き換えたものとする（**Fig. 1**）．ニューラルネット自体は，通常の階層型ニューラルネットである．ここでは，3層のニューラルネットで説明を行なう．

扱う単語（入力記号）の全体集合を$\{W_1, W_2, \cdots, W_M\}$とし，単語と$d$次元実ベクトルの対応を示した記号表現ベクトル辞書を$\boldsymbol{L} = [r_{11}, r_{12}, \cdots, r_{1d}, r_{21}, \cdots, r_{Md-1}, r_{Md}]$とする．記号表現ベクトル$\boldsymbol{r}_i = [r_{i1}, r_{i2}, \cdots, r_{id}]$は$W_i$に対応する実ベクトルである．入力単語列を$W_{\sigma(1)}, W_{\sigma(2)}, \cdots, W_{\sigma(m)}$ $(1 \leq \sigma(i) \leq M)$とすると，これに対応する階層型ニューラルネットへの入力パターンは，各単語の記号表現ベクトル$\boldsymbol{r}_{\sigma(1)}, \boldsymbol{r}_{\sigma(2)}, \cdots, \boldsymbol{r}_{\sigma(m)}$をこの順に連結したベクトルである．つまり，$\boldsymbol{r}_{\sigma(k)}$の第$j$成分が，入力層の第$(k-1)d+j$ユニットの値として与えら

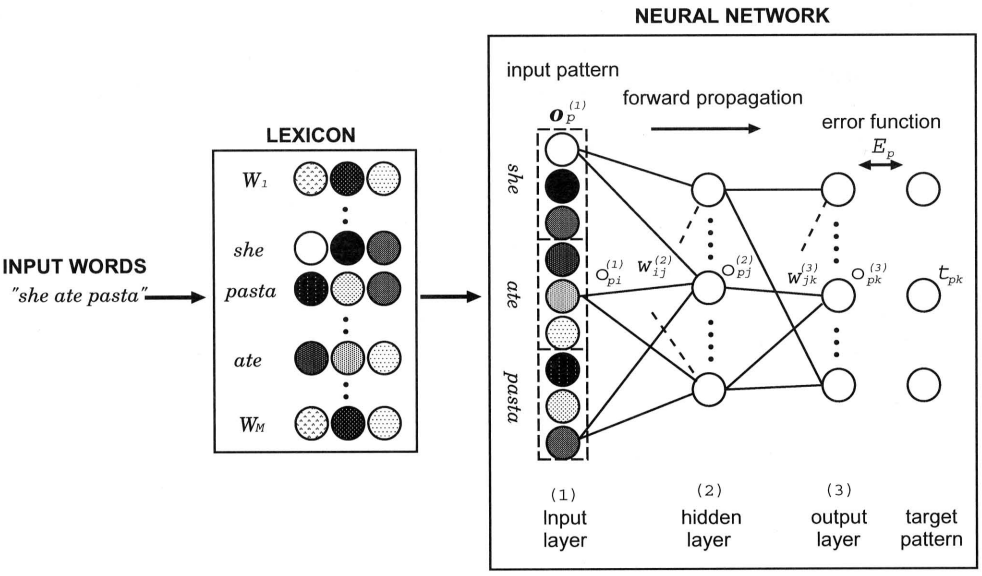

Fig. 1　Structure of the layered neural network used in this paper.

れる．

以下では，入力側から数えてl番目の層における第jユニットの出力値を$o_j^{(l)}$と表記する．$o_j^{(l)}$は，

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

で計算される．ここで，$w_{ij}^{(l)}$は第$(l-1)$層第iユニットと第l層第jユニットとの結合荷重，$b_j^{(l)}$は第l層第jユニットのバイアス値と呼ばれ，これら$w_{ij}^{(l)}$，$b_j^{(l)}$は学習パラメタである．全$w_{ij}^{(l)}$，$b_j^{(l)}$を集めたベクトルを$\boldsymbol{w}$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．また，n_lは第l層のユニット数を表わす．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．以上のようにして，ネットワークの出力値が求まる．

学習パラメタである結合荷重$\boldsymbol{w}$の学習は，入力文s_pとそれに対する望ましい出力である教師信号（target pattern）$\boldsymbol{t}_p$の組の列$(s_1, \boldsymbol{t}_1)$ $(s_2, \boldsymbol{t}_2)$ $\cdots$ $(s_N, \boldsymbol{t}_N)$を学習データとし，ニューラルネットの出力と教師信号との平均二乗誤差

$$E(\boldsymbol{w}) = \frac{1}{N}\sum_{p=1}^{N} E_p = \frac{1}{N}\sum_{p=1}^{N}\sum_{k=1}^{n_3} \frac{1}{2}(o_{pk}^{(3)} - t_{pk})^2$$

を目的関数とし，最急降下法により$E(\boldsymbol{w})$を最小（実際には極小）にするように$\boldsymbol{w}$を求めることで行なわれる．ここで，1/2は計算の便宜上導入される定数で，t_{pk}は$\boldsymbol{t}_p$の第k成分，$o_{pk}^{(l)}$は入力文としてs_pが与えられたときの第l層第kユニットの出力値である．また以下では，入力文としてs_pが与えられたときの第l層第kユニットの内部ポテンシャルを$u_{pk}^{(l)}$で示す．

目的関数$E(\boldsymbol{w})$の値を極小にする学習は，誤差逆伝搬法（BP法）と呼ばれており，ニューラルネットの学習において，よく知られている[2]．

2.2　記号表現ベクトル学習型ニューラルネットの学習

記号表現ベクトル学習型ニューラルネットの学習の構成は**Fig. 1**と同一である．目的関数は次式で示される．

$$E(\boldsymbol{w}, \boldsymbol{L}) = \frac{1}{N}\sum_{p=1}^{N}\sum_{k=1}^{n_3} \frac{1}{2}(o_{pk}^{(3)}(\boldsymbol{w}, \boldsymbol{L}) - t_{pk})^2.$$

本手法の学習については，$\boldsymbol{w}$と$\boldsymbol{L}$の双方を学習パラメタとみなし，目的関数$E(\boldsymbol{w}, \boldsymbol{L})$の値を極小にする学習を考える．この場合，$\nabla_w E(\boldsymbol{w}, \boldsymbol{L}) \neq \mathbf{o}$，または，$\nabla_L E(\boldsymbol{w}, \boldsymbol{L}) \neq \mathbf{o}$であるならば，ある$\eta(>0)$が存在して，

$$E(\boldsymbol{w}', \boldsymbol{L}') < E(\boldsymbol{w}, \boldsymbol{L})$$
$$\boldsymbol{w}' = \boldsymbol{w} - \eta\nabla_w E(\boldsymbol{w}, \boldsymbol{L})$$
$$\boldsymbol{L}' = \boldsymbol{L} - \eta\nabla_L E(\boldsymbol{w}, \boldsymbol{L})$$

が成立することを利用して，

$$w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \eta\frac{\partial E}{\partial w_{ij}^{(l)}},$$
$$r_{ce} \leftarrow r_{ce} - \eta\frac{\partial E}{\partial r_{ce}}$$

のようにパラメタを順次更新することで$E(\boldsymbol{w}, \boldsymbol{L})$を極小にする$\boldsymbol{w}$と$\boldsymbol{L}$を求めることができる．なお学習において，結合荷重の値には制限を設けないが，記号表現ベクトル

の各要素は$[-1.0, +1.0]$に収まるようにする†1．

$\partial E/\partial w_{ij}^{(l)}$については通常のBP法と同様である．$\partial E/\partial r_{ce}$については，

$$\begin{aligned}\delta_{pi}^{(1)} &= \frac{\partial E_p}{\partial o_{pi}^{(1)}} \\ &= \sum_{j=1}^{n_2} \frac{\partial E_p}{\partial u_{pj}^{(2)}} \frac{\partial u_{pj}^{(2)}}{\partial o_{pi}^{(1)}} \\ &= \sum_{j=1}^{n_2} \delta_{pj}^{(2)} w_{ij}^{(2)}\end{aligned}$$

とおくと，

$$\begin{aligned}\frac{\partial E}{\partial r_{ce}} &= \frac{1}{N} \sum_{p=1}^{N} \sum_{i=1}^{n_1} \frac{\partial E_p}{\partial o_{pi}^{(1)}} \frac{\partial o_{pi}^{(1)}}{\partial r_{ce}} \\ &= \frac{1}{N} \sum_{p=1}^{N} \sum_{i=1}^{n_1} \delta_{pi}^{(1)} \frac{\partial o_{pi}^{(1)}}{\partial r_{ce}}\end{aligned}$$

となる．ただし，

$$\begin{aligned}\delta_{pj}^{(2)} &= \frac{\partial E_p}{\partial u_{pj}^{(2)}} \\ &= \sum_{k=1}^{n_3} \frac{\partial E_p}{\partial u_{pk}^{(3)}} \frac{\partial u_{pk}^{(3)}}{\partial o_{pj}^{(2)}} \frac{\partial o_{pj}^{(2)}}{\partial u_{pj}^{(2)}} \\ &= \sum_{k=1}^{n_3} \delta_{pk}^{(3)} w_{jk}^{(3)} \frac{\partial o_{pj}^{(2)}}{\partial u_{pj}^{(2)}} \\ &= f'(u_{pj}^{(2)}) \sum_{k=1}^{n_3} \delta_{pk}^{(3)} w_{jk}^{(3)}\end{aligned}$$

であり，

$$\begin{aligned}\delta_{pk}^{(3)} &= \frac{\partial E_p}{\partial u_{pk}^{(3)}} \\ &= \frac{\partial E_p}{\partial o_{pk}^{(3)}} \frac{\partial o_{pk}^{(3)}}{\partial u_{pk}^{(3)}} \\ &= f'(u_{pk}^{(3)})(o_{pk}^{(3)} - t_{pk})\end{aligned}$$

である．

2.3 PP-attachmentのためのモデル

本稿では，前置詞句の係り先が

1. 他動詞V
2. 名詞句1の主辞の名詞N_1
3. 前置詞P
4. 名詞句2の主辞の名詞N_2

という四つ組$< V, N_1, P, N_2 >$のみから決定できるものと仮定し，学習データもテストデータもこの四つ組で表わす．このため入力層を4つのユニット群に分け，それぞれにV, N_1, P, N_2に対応する記号表現ベクトルを入力する．本手法をPP-attahchment問題に適用したニューラルネットの構成を**Fig. 2**に示す．出力層は1ユニットで，教師信号としては，前置詞句が名詞N_1に係る場合は0.0を，動詞Vに係る場合は1.0をそれぞれ与える．学習終了後のテストにおいては，出力値が閾値$\theta(= 0.5)$より小さければ名詞N_1に，θ以上であれば動詞Vに係ると判定する．

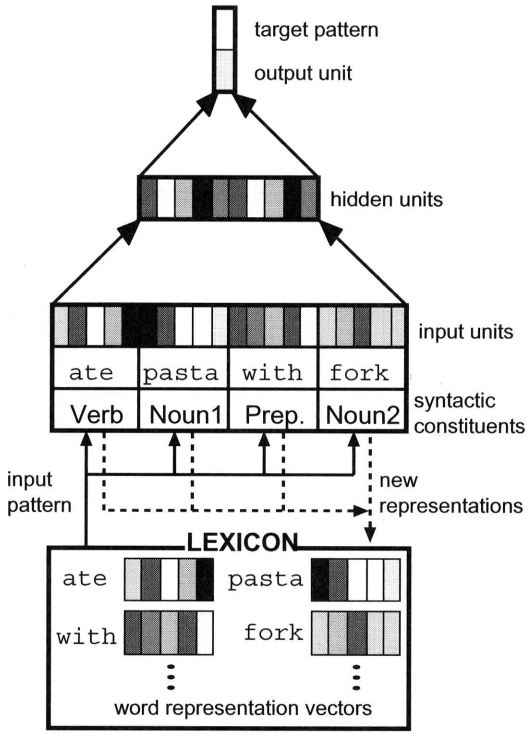

Fig. 2 Structure of the proposed method for PP-attachment.

3. 実　　験

本章では，本手法によるPP-attachment問題について，以下の2つの計算機実験を行ない評価する．

実験1：固定表現を用いた通常のBP法と，記号表現ベクトル辞書$\boldsymbol{L}$も学習する本手法との比較

実験2：中間ユニット数を増やすことでパラメタ数を増やした通常のBP法と，記号表現ベクトル辞書$\boldsymbol{L}$もパラメタとする本手法との比較

記号表現ベクトル辞書$\boldsymbol{L}$を学習することで問題に適した$\boldsymbol{L}$を獲得できているならば，単語列としては未知の（つまり学習データに現れない）データに対してもある程度高い性能を示すことが期待できる．これが$\boldsymbol{L}$を学習することの効果である．そこで，実験1，2では未知のデータに対する性能（以降，汎化性能と呼ぶ）により，本手法と通常のBP法とを比較することで記号表現ベクトル辞書$\boldsymbol{L}$を学習パラメタとすることの効果を探る．

3.1 実験データ

実験に使用したデータはRatnaparkhiらがインターネット上に公開しているTreebankコーパスの四つ組デー

†1　記号表現ベクトルは，(1)式の更新で収束しない（各要素が$+\infty$，または$-\infty$になる）可能性がある．この制限はこれを防ぐためである．文献 3) でもそのような制限を設けている．

タ[4]（学習データ20801個，テストデータ3097個，メタパラメタ決定用データ4039個）を元に作成した．他の先行研究で行なっているように，データスパースネスに対応するため，実験データの形態素に対する前処理を以下のように施した†2．

- 全ての数字を疑似単語 'num'で置き換えた．
- 全ての大文字を小文字に変換した．
- 全ての動詞と全ての名詞は形態素解析器XTAG[5]を用いて原形に変換した．

また，同一のスペルで名詞と動詞の両方に用いられている単語は異なる2語とし，名詞用と動詞用の2つの記号表現ベクトルを用意した．

結果として，実験に用いた四つ組データ数，単語数は各実験毎に異なる．データ数，単語数は以下の各節にて説明する．

メタパラメタ決定用データは，モデルを評価するための学習回数を決めるために用いた．以下の実験では，メタパラメタ決定用データに対する正解率が最高値をとる学習回数における，テストデータに対する正解率をモデルの汎化性能として評価した．テストデータを用いて学習回数を決めないのは，評価がテストデータに特化する恐れを回避するためである．

3.2 実験1

本節では，数種類の初期$\boldsymbol{L}$（記号表現ベクトル辞書）に対して，初期$\boldsymbol{L}$に固定した通常のBP法と，$\boldsymbol{L}$を学習する本手法との比較実験について述べる．

3.2.1 実験条件

記号表現ベクトル辞書$\boldsymbol{L}$として，シソーラスに基づく表現（比較的問題に適していると考えられる表現）$\boldsymbol{L}_0$と，2種類のランダムベクトル$\boldsymbol{L}_1$，$\boldsymbol{L}_2$の計3種類を用意する．通常のBP法は$\boldsymbol{L}$を$\boldsymbol{L}_0$〜$\boldsymbol{L}_2$に固定し，$\boldsymbol{w}$のみを学習する．一方，本手法は$\boldsymbol{L}_0$〜$\boldsymbol{L}_2$を初期値として$\boldsymbol{w}$および$\boldsymbol{L}$を学習する．$\boldsymbol{w}$の初期値は両手法ともに，$(-1.0, 1.0)$上の一様乱数に基づき設定する．ネットワークの中間ユニット数は50，記号表現ベクトルの次元数は7とした．

シソーラスに基づく表現（$\boldsymbol{L}_0$）は基本的には文献 1)の方法と同様にして作成した．すなわち，名詞と動詞に対しては，WordNet1.6[6]を用いて類似度行列を作成し，類似度行列から数量化理論IV類†3 を用いて記号表現ベクトルを求めた．前置詞はWordNetに含まれないので，各要素を$(-1.0, 1.0)$上の一様乱数とした．

また，$\boldsymbol{L}_1$，$\boldsymbol{L}_2$は以下のようにして設定した．名詞と動詞に関しては，

$\boldsymbol{L}_1$: 各要素が$(-1.0, 1.0)$上の一様乱数

$\boldsymbol{L}_2$: 各要素が$(-0.001, 0.001)$上の一様乱数

とした．前置詞に関しては，$\boldsymbol{L}_0$と同様，$\boldsymbol{L}_1$，$\boldsymbol{L}_2$に対して，各要素を$(-1.0, 1.0)$上の一様乱数とした．

以上説明した実験条件を**Table 1**に，使用した四つ組データ数および単語数を**Table 2**，**Table 3**に示す．

Table 1 Conditions of the experiment 1.

Model	BP	Proposed method
Learning of $\boldsymbol{L}$	off	on
# of dim. of Vector	7	
Initial values of $\boldsymbol{L}$	$\boldsymbol{L}_0 \sim \boldsymbol{L}_2$	
# of hidden units	50	
Initial value of $\boldsymbol{w}$	random of $(-1.0, 1.0)$	
η	1.0×10^{-4}	
Momentum term	none†4	

Table 2 Number of quadruples for the experiment 1.

Quadraples	Training	Test	Development
# of data	14733	1828	2210

Table 3 Number of words for the experiment 1.

	Noun	Verb	Preposition
Each word	4366	1540	66
Total word	37542	18771	18771

3.2.2 結果および考察

各$\boldsymbol{L}$の場合における，汎化性能を**Table 4**にまとめた．参考までに，メタパラメタ決定用データに対する正解率が最高値をとる学習回数における学習データに対する正解率を**Table 5**に示す．

Table 4より以下のことがいえる．

1. 全ての$\boldsymbol{L}$に対して，本手法の方が，通常のBP法に比べて高い性能を示す．このことから，$\boldsymbol{L}$を学習する本手法が，問題に適した$\boldsymbol{L}$を獲得している．
2. $\boldsymbol{L}$を$\boldsymbol{L}_0$に固定した通常のBP法での汎化性能より，$\boldsymbol{L}_1$あるいは$\boldsymbol{L}_2$を初期$\boldsymbol{L}$として本手法により$\boldsymbol{L}$を学習させたときの汎化性能の方が高い．このことから，シソーラスのような外部知識を用いて$\boldsymbol{L}$を構成するよりも，ランダムに初期$\boldsymbol{L}$を構成して本手法により$\boldsymbol{L}$を学習させる方が，問題に適した$\boldsymbol{L}$を獲得できる．
3. 本手法により$\boldsymbol{L}$を学習させたとする．$\boldsymbol{L}_2$を初期$\boldsymbol{L}$とした場合の汎化性能は，$\boldsymbol{L}_0$を初期$\boldsymbol{L}$とした場合より若干低いが，大きな違いはない．本手法を用いる場合の初期$\boldsymbol{L}$として，シソーラスのような外部知識を用いて$\boldsymbol{L}$を構成しなくても，ランダムに設定した初期$\boldsymbol{L}$をいくつか試せば，ほぼ同程度に問題に適した$\boldsymbol{L}$を獲得できる．

†2 前処理の方法は，各研究者によって異なり，いくつかの方法がなされているが，今回はRatnaparkhiらと全く同じ方法を用いた[7]．

†3 数量化理論IV類は類似度行列から各個体をユークリッド空間に配置する方法の一つである[8]．数量化理論IV類では，単語間の類似度が高い単語同士の記号表現ベクトルは互いに近く，単語間の類似度が低い単語同士の記号表現ベクトルは互いに遠くになるように決定される．

†4 本稿の実験は，Eの極小値への収束を目標とするので，慣性項は設定しない．

Table 4 Maximum performance for the test data.

$\boldsymbol{L}$(initial value)	BP(%)	Proposed mothod(%)
$\boldsymbol{L}_0$	73.5	83.3
$\boldsymbol{L}_1$	76.1	77.2
$\boldsymbol{L}_2$	72.3	82.9

Table 5 Performance for the training data.

$\boldsymbol{L}$(initial value)	BP(%)	Proposed method(%)
$\boldsymbol{L}_0$	74.3	89.5
$\boldsymbol{L}_1$	81.4	92.0
$\boldsymbol{L}_2$	72.4	90.0

シソーラスを基にした表現$\boldsymbol{L}_0$は，一見，ランダムよりは良さそうに思える表現である．しかし今回の実験では，通常のBP法に対して，シソーラスに基づいた$\boldsymbol{L}$（$\boldsymbol{L}_0$）の方が，ランダムに定めた$\boldsymbol{L}$（$\boldsymbol{L}_1$）よりも性能が低いという結果を得た．これは，記号表現ベクトルの次元数が少なかったためであると考えられる．しかし，記号表現ベクトルの次元数を多くしていくと，過学習が発生しやすくなる．過学習を防ぐためには，膨大な数の学習データを必要とする．このように，問題に適した表現（の初期値）を設定するのは困難である．その点，本手法であれば，ランダムに定めた初期$\boldsymbol{L}$を基にして，ある程度問題に適した記号表現ベクトル辞書$\boldsymbol{L}$が学習できると考えられる．

3.3 実験2

パラメタ$\boldsymbol{w}$（結合荷重）は，入力となる四つ組データ全体の学習に関係するのに対し，パラメタ$\boldsymbol{L}$（記号表現辞書）は，各記号表現ベクトルがごく一部の四つ組の学習にしか関係していない．つまり，これら2種類のパラメタはそれぞれ性質が異なり，全く同じ取り扱いはできないと考えられる．しかし，$\boldsymbol{L}$の学習を許すことは，結局はパラメタを増やす（システムの自由度を高くする）ことにつながる．実験1では，$\boldsymbol{L}$を学習させたときの方が，$\boldsymbol{L}$を固定したときよりも高い正解率を示したが，これは$\boldsymbol{L}$の学習を許すことが本質的な理由ではなく，単にパラメタ数が増大したからかもしれない．パラメタ数を増やすだけならば，中間ユニット数を増やすことでも可能である．本節では，通常のBP法で単に中間ユニット数n_2を 50, 100, 150, 200 と増やしたときの効果と，$n_2 = 50$ で$\boldsymbol{L}$の学習を許したとき（つまり本手法）の効果との相違を，実験的に示す．

3.3.1 実験条件

記号表現ベクトルの次元数は7とし，各パラメタの初期値は$(-1.0, 1.0)$のランダムとした．

最急降下法によりEを（最小ではなく）極小にするパラメタ値を求めているため，当然，汎化性能もパラメタの初期値に依存する．そこで，パラメタの初期値の与え方を5通り試した．中間ユニット数$n_2 = 50$に対し，$(-1.0, 1.0)$上の一様乱数に基づく$(\boldsymbol{w}, \boldsymbol{L})$を5通り試し，各初期値に対して通常のBP法による学習を行ない，このうち，最も良い汎化性能を求めた．これと同様の実験を中間ユニット数が100, 150, 200の各場合についても行なった．また，本手法に対する$(\boldsymbol{w}, \boldsymbol{L})$の初期値は，通常のBP法（中間ユニット数 50）で最良の汎化性能を示したときに与えた$(\boldsymbol{w}, \boldsymbol{L})$の初期値と同じものを用いた．通常のBP法で最良のパラメタの初期値が，本手法に対して最良の初期値とは限らないので，本手法にとっては控えめな評価法といえる．

Table 6 Conditions of the experiment 2.

	BP	Proposed method
Learning of $\boldsymbol{L}$	off	on
# of dim. vector	7	
# of hidden units	50, 100, 150, 200	50
η	1.0×10^{-4}	
Momentum term	none	
Initial value of $\boldsymbol{L}$	random of $(-1.0, 1.0)$ †5	
Initial value of $\boldsymbol{w}$	random of $(-1.0, 1.0)$ †5	

Table 7 Number of quadruples for the experiment 2.

Quadraple	Training	Test	Development
# of data	20801	2442	2870

Table 8 Number of words for the experiment 2.

	Noun	Verb	Preposition
Each word	5582	1674	66
Total word	52226	26113	26113

Table 9 Generalization abilities of the standard BP and the proposed method.

# of hidden unit n_2	Generalization ability (%)	
	BP	Proposed method
50	75.5	79.4
100	76.6	
150	76.5	
200	76.9	

以上の実験条件を**Table 6**に，使用した四つ組データ数および単語数を**Table 7**，**Table 8**に示す．

3.3.2 結果および考察

Table 9に，テストデータに対する正解率をまとめた．通常のBP法において中間ユニット数を200まで増やしても，本手法における中間ユニット数が50のときの正解率79.4%に比べて，良い正解率を示すことはなかった．

本実験より，通常のBP法と本手法では汎化性能に違いがあることがわかるが，これは以下のような汎化の原理の違いからくるものであると考えられる．

通常，汎化とはシステムに対して未知の入力に対する処理をいう．この問題の場合，学習データには現れない新たな単語列に対して，学習データから獲得した処理知識を適用することになる．

正しい汎化は，中間層と出力層により学習データに応じて分割された入力特徴ベクトル空間のクラス領域に，テストデータの単語列を入力ベクトルに変換したベクト

†5 5通り試した中で通常のBP法に対して最良のもの

ルが，適正に配置された時になされる．つまり，学習データの単語列の入力ベクトルとテストデータの単語列の入力ベクトルとが互い近い時に正解となる．

例えば，今，文s_lを含みかつ文s_tを含まないような学習データを使ってモデルを学習させたとする（教師信号はVで同一）．

s_l: eat pasta with fork

s_t: taste soup with spoon

学習データにはs_l以外にも多くの文が含まれている．そこでは eat と taste, pasta と soup, fork と spoonは比較的類似した使われかたがなされていると考えられる．本手法では，類似した使われかたがなされている単語の記号表現ベクトルは互いに近いベクトルとなるので，eat と taste，pasta と soup，fork と spoon の記号表現ベクトルはそれぞれ互いに近いベクトルとなっているはずである．したがって，文s_lの入力ベクトルと文s_tの入力ベクトルも互いに近くなるので，未知の文s_tは正解と判断される可能性が高くなる．以上のように，記号表現ベクトルの学習は，入力特徴ベクトル空間におけるテストデータの入力ベクトルの配置を間接的に行ない，汎化に役立っているといえる．

一方，記号表現ベクトルの学習を行なわない場合は，eat と taste，pasta と soup，fork と spoon の記号表現ベクトルが必ずしも互いに近いベクトルになっているとは限らない．このため，文s_tの入力ベクトルが，s_lの入力ベクトルより学習データ中の他の文

s_l': taste soup with caviar

（ただし，教師信号はN）の入力ベクトルに近いというようなこともあり，汎化に失敗する場合が多いと考えられる．

パラメタ数を増やす場合，通常のBP法において中間ユニット数を増やした時は，入力特徴空間において分割されるクラス領域が増えるだけで，必ずしもよりよい汎化がなされる保証はない．一方，本手法にて記号表現ベクトルも学習パラメタとした時は，通常のBP法の汎化の原理に上記のような合理的な汎化の原理が加わるため，通常のBP法よりも少ない中間ユニット数でより高い汎化性能が得られたと考えられる．

4. おわりに

本稿では，記号表現ベクトル学習型ニューラルネットとその学習法を示した．さらに，本手法の有効性を実験的に示すために，本手法をPP-attachment問題に適用した実験を行なった．これにより，

1. 固定された記号表現ベクトルを用いる通常のBP法より本手法の方が汎化性能が高いこと
2. 記号表現ベクトルを学習することと，単に中間ユニット数を増やすこととには汎化性能に違いがあること

を示した．1に関しては，何種類もの固定表現を試したわけではないが，少なくとも性能が低くなるという結果は見られなかった．2に関しては，パラメタの初期値を変えての試行回数が十分でないため，記号表現ベクトルを学習した場合は，学習しない場合よりも必ず汎化性能が高いとは断言できない．今後，さらにパラメタの初期値を変えて実験する必要がある．

なお，本手法は，第1層と第2層とが全結合でない特殊な4層ニューラルネットに相当する．しかし，このようなニューラルネットが通常の学習で実現できるとは考えにくく，しかも本手法に比べ非効率である．ただし，パラメタを増やす方法として中間層数を増やすことも，試してみるべき課題である．

参考文献

1) Naoto Takahashi, Minoru Motoki, Yoshio Shimazu, Yoichi Tomiura, and Toru Hitaka. Pp-attachment ambiguity resolution using a neural network with modified fgrep method. In *Proceedings of the Second Workshop on Natural Language Processing and Neural Networks*, pp. 1–7, 2001.
2) D.E. Rumellhart, G.E. Hinton, and R.J. Williams. *Learning Internal Representations by Error Propagation*, Vol. 1. The MIT Press, Massachusetts, 1986.
3) R. Miikkulainen and M. G. Dyer. Natural language processing with modular pdp networks and distributed lexicon. *Cognitive Science*, Vol. 15, pp. 343–399, 1991.
4) Michael Collins and A Ratnaparkhi. ftp://ftp.cis.upenn.edu/pub/adwait/PPattachData/.
5) D. Karp, Y. Schabes, M. Zaidel, and D. Egedi. A freely available wide coverage morphological analyzer for english. In *Proceedings of COLING '92*, pp. 950–954, 1992.
6) George A. Miller. *Introduction to WordNet: An On-line Lexical Database*. http://www.cogsci.princeton.edu/~wn/, 1993.
7) A Ratnaparkhi. Maximum entropy models for natural language ambiguity resolution. In *Ph.D. Dissertation, University of Pennsylvania*, 1998.
8) 林知己夫. 数量化–理論と方法–. 朝倉書店, 1993.