ニューラルネットとルールベース手法を統合した品詞タグづけシステム

馬 青[†] 内元清貴[†]
村田真樹[†] 井佐原 均[†]

著者らはこれまで，できるだけ少量の訓練データで実用的な品詞タグづけシステムを構築する目的で情報量最大を考慮し最長文脈を優先するニューロタガーを提案してきた．すでに提案したタガーは数万オーダーの多品詞語を含む小規模タイ語コーパスを訓練に用いることにより，テストデータを94.4%の精度（多品詞語のみを測定対象）でタグづけすることができた．この精度はタグづけの主要手法とされてきた隠れマルコフモデルのそれ（89.1%）より遥かに高く，誤り駆動学習で得られた書き換え規則のみで構成されるルールベースモデルのそれ（93.5%）よりも高かった．本論文は，品詞タグづけ性能を更に改善するために，ニューロタガーの弱点を補う書き換え規則を誤り駆動学習で自動獲得して後処理に導入し，ニューラルネットとルールベースを統合するシステムを提案する．書き換え規則を後処理に用いることによりタグづけのエラーは19.7%減少し，全体の統合システムのタグづけ精度は95.5%まで向上した．

An Integrated Neuro and Rule-Based Part of Speech Tagger

Qing Ma ,[†] Kiyotaka Uchimoto ,[†] Masaki Murata [†]
and Hitoshi Isahara[†]

So far, the neuro part of speech tagger that uses different lengths of contexts based on longest context priority and takes into account the maximization of information amount have been proposed for the purpose of constructing a practical tagger which uses as few training data as possible. The proposed neuro tagger has tagging accuracy of 94.4% (for accounting only the ambiguous words in part of speech), when a small Thai corpus with ten thousand order words was used for training. This accuracy is far higher than that of using Hidden Markov Model, a main approach for part of speech tagging, and is also higher than that (93.5%) of using a transformation rule based model. To further improve the tagging performance, this paper proposes an integrated neuro and rule-based tagger by introducing a set of transformation-based rules as a post-processor of the neuro tagger. The rules are neuro tagger oriented and are acquired by the error driven learning. The 19.7% of errors made by the neuro tagger can be corrected by rules and the whole tagger reaches an accuracy of 95.5%.

1. はじめに

日本語や中国語，タイ語などわかちがきされていない言語を解析するとき，まずそれを語単位（それを形態素と呼ぶ）に，それらの属性（品詞，活用など）を解析しながら正しく分解しなければならない．これは形態素解析と言い，「構文解析」「意味解析」「文脈解析」と合わせ，自然言語処理の四つの基本解析とされている[1),2)]．形態素解析においては語の種々の属性が重要な役割を演じるが，活用形や活用型といった属性は曖昧性が少なく語彙辞書を調べれば済むのに対し，品詞という属性は文脈によって変わるためそう簡単には決められない．従って，形態素解析では多品詞語の曖昧性を文脈によって解消すること（品詞タグづけ）が重要な課題である．実際，品詞タグづけは形態素解析や構文解析，更には機械翻訳といった自然言語処理に欠かせない基本技術だけでなく，その技術は，さらに，音声合成の前処理，OCR や音声認識の後処理，そして情報検索などへの幅広い応用も考えられる．

品詞タグづけに関する研究は自然言語処理研究を始めた頃から行なわれてきており，ルールベースや統計手法を主流とする数多くの品詞タグづけシステム（例えば，ルールベース[3)〜5)]，確率モデル[6)〜13)]，ニューラルネット[14)〜16)]，最大エントロピー法[17)]，決定木[18),19)]）が提案されてきた．それらのほとんどはタグづけに長さが固定の文脈を用いるものであり（隠れマルコフモデル（HMM）においても状態遷移を定

† 郵政省通信総合研究所
Communications Research Laboratory, Ministry of Posts and Telecommunications, Japan

義するのに固定された n-gram ベースのモデルを用いる），入力の各構成部分は同一の影響度を持つものとされていた．それらのシステムで高い精度を出すためには大量（英語の場合 1,000,000 オーダ）の訓練データを用いることを前提とする必要がある．しかしながら，実際，英語や日本語などを除いた数多くの言語（例えば本稿で取り上げたタイ語）に関しては，コーパス自体もまだ整備段階にあるのが現状で，予め大量の訓練データを得るのが困難である．

そのために，筆者らは少ない訓練データを用いても，十分実用的な品詞タグづけが可能なマルチニューロタガー[20),21)]，更にはそれを大幅にスリム化した伸縮性ニューロタガー[22)]を提案してきた．提案したニューロタガーは主に以下の二つの特徴を有する：(1)タグづけ結果の確信度を高めるために，タグづけはまずできるだけ長い文脈を用いて行ない，訓練データの不足から確定的な答えが出ない場合に順次文脈を短くするといったように長さ可変文脈を用いている；(2)タグづけにおいては，目標単語自身の影響が最も強く，前後の単語もそれぞれの位置に応じた影響度を持つことを反映させるために，入力の各構成部分は情報量最大を考慮して訓練データから得られるインフォメーションゲイン（略して IG と呼ぶ）[23),24)]を影響度として重み付けられている．このようなニューロタガーは，数万オーダーの多品詞語を含む小規模タイ語コーパスを訓練に用いた場合，HMM（89.1%）より遥かに高い精度（94.4%）でテストデータ（多品詞語のみ）を品詞タグづけできた[☆1]．

しかしながら，ニューラルネットを用いた手法は統計的手法と同様「統計的」に解析を行なうもので「確実」な規則を取り扱うことが困難である．例えばある単語の品詞が前の単語のみによって「確実」に決まると仮定しよう．この場合でも，ニューラルネットはあくまでも文脈全体の下での可能性に基づいて「統計的」に解析を行なう．その結果，前の単語が同じでも全体の文脈が変わると，タグづけ結果が変わってしまう可能性がある．また，単語そのものが品詞タグづけに非常に重要な情報であるにもかかわらず，ニューラルネットはその規模が膨大になることから単語そのものを入力に用いるのが困難である．更に，ニューラルネットはその収束性と過学習の問題から訓練データへの品詞タグづけ精度を 100%まで学習し続けるのはほとんど不可能であり得策ではないことから，適当と思われる精度で学習を止めるのが普通である．そのため，有用な規則が獲得されていない恐れがある．

本研究では，ニューラルネット手法のもつこのような弱点を補うために書き換え規則を後処理[☆2]に導入し，ニューラルネットとルールベースの統合システムを構築[☆3]した．ここで用いるニューラルネットは，単一のニューラルネットでありながら，可変長の入力に対応するように構成した伸縮性ニューロタガー[22)]である．ニューロタガーの後処理に用いた書き換え規則は，単語そのものの情報を活用するなどニューラルネットの弱点を補うテンプレートを用いて誤り駆動型学習[5)]により訓練データから自動獲得される．計算機実験の結果，書き換え規則を後処理に用いることによってテストデータへのニューロタガーによる品詞タグづけのエラーは 19.7%減らされ，全体の統合システムのタグづけ精度はニューロタガーのみを用いた場合より 1.1%高く，95.5%まで向上した．

2. 品詞タグづけ問題

本研究では，単語の取りうる品詞がリストアップされている辞書

$$V = (w^1, w^2, \cdots, w^v) \qquad (1)$$

と品詞セット

$$\Gamma = (\tau^1, \tau^2, \cdots, \tau^\gamma) \qquad (2)$$

が既に用意されているものと仮定する（但し，v は登録された単語の総数で，γ は品詞の数である）．つまり，本研究では（辞書に存在しない）未知語は取り扱わない[☆4]．従って，品詞タグづけ問題は，任意の文 $W = w_1 w_2 \cdots w_s$（$w_i \in V,\ i = 1, \cdots, s$）が与えられた時，以下の手続き φ によって品詞列 $T = \tau_1 \tau_2 \cdots \tau_s$（$\tau_i \in \Gamma,\ i = 1, \cdots, s$）を見つけることである．

$$\varphi : W^t \to \tau_t,\ \ t = 1, \cdots, s \qquad (3)$$

但し，t は品詞を定めようとする目標単語のインデックスを表し，W^t は目標単語 w_t を中心とした長さ $l+1+r$ の単語列である．即ち，

$$W^t = w_{t-l} \cdots w_t \cdots w_{t+r} \qquad (4)$$

☆1 これまでの文献では品詞タグづけの精度としてよくテストデータの全単語（品詞の曖昧性のありなし問わず）を対象に測定して得たものを用いている．もしこのような定義を用いれば，ニューロタガーの精度は 98.9%にも達した．

☆2 書き換え規則を後処理に用いる考え方は Brill[5)] によって初めて提案されたもので，日本語の形態素処理においては最近ルールベース手法の後処理に書き換え規則を用いる久光ら[25)]の研究がある．

☆3 ルールベース手法を導入して構築した統合システムは前述したニューラルネット手法の弱点を補うための一つの解決方法であり，多数の手法から考えられるさまざまな組み合わせ（例えば HMM と書き換え規則など）の中でもっとも適切なものである保証はない．

☆4 本論文で実験に用いた学習用およびテスト用のコーパスは，人手でタグ付けをされたものであり，いわゆる未知語は存在しない．

但し，$t-l \geq 1,\ t+r \leq s$．従って，品詞タグづけは品詞をクラスに置き換えたクラス分け問題としてとらえることができ，ニューラルネットで取り扱うことができる．

3. 統合システム

提案する統合システム（図１）は主タグづけ器としてのニューロタガーと後処理としてのルールベースコレクターから構成される．文が入力されると，まずニューロタガーが個々の単語に品詞タグづけを行なう．そして，ルールベースコレクターが微調整のチューナとしてニューロタガーの出力を必要に応じて修正し，最終的に品詞を決定する．以下，ニューロタガーとルールベースコレクターについてそれぞれ詳しく述べる．

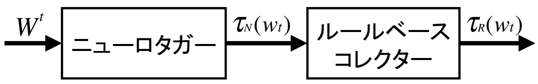

図 1 統合システム
Fig. 1 The integrated system

3.1 ニューロタガー

ニューロタガーは，単一の三層パーセプトロン（図２）で構成される．但し，ニューロタガーはその入力が伸縮可能なものとされているため，品詞タグづけを長さ可変文脈で行うことができる．この節では，順に入出力構成，入力の重みづけに用いるインフォメーションゲイン，そして伸縮性入力に適した訓練方法について述べる．三層パーセプトロンのアーキテクチャやニューロタガーの特徴などについては文献21)，22)を参照されたい．

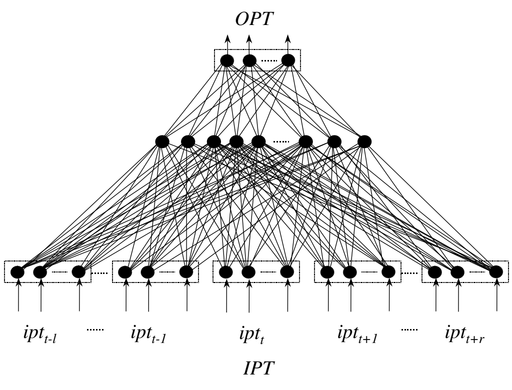

図 2 ニューロタガー
Fig. 2 The neuro tagger

3.1.1 入出力構成

入力 IPT は目標単語 w_t を中心とした長さ $l+1+r$ の単語列 W^t[式 (4)] から得られた情報で構成されるもので，以下のように表す．

$$IPT = (ipt_{t-l}, \cdots, ipt_t, \cdots, ipt_{t+r}) \quad (5)$$

但し，入力の長さ $l+1+r$ は固定なものではなく，後で述べるように伸縮性をもつものとされる．具体的に，単語 w が入力の位置 x $(x = t-l, \cdots, t+r)$ に与えられた時，IPT の構成部分である ipt_x は以下のように重み付けされたパターンで定義される．

$$ipt_x = g_x \cdot (e_{w1}, e_{w2}, \cdots, e_{w\gamma}) \quad (6)$$

ここで，g_x は次節で述べるインフォメーションゲインで [(16)-(19) を用いて] 求められる重みで，γ は品詞の数である．もし単語 w が訓練データに出現するならば，各要素 e_{wi} は以下のように得られる．

$$e_{wi} = Prob(\tau^i|w) \quad (7)$$

但し，$Prob(\tau^i|w)$ は単語 w の品詞が τ^i である確率で，訓練データから以下のように推定される．

$$Prob(\tau^i|w) = \frac{C(\tau^i, w)}{C(w)} \quad (8)$$

ここで，$C(\tau^i, w)$ は全訓練データを通じ，w が品詞 τ^i を取る回数で，$C(w)$ は w が出現する回数である．一方，もし単語 w が訓練データに出現しないならば，各要素 e_{wi} は以下のように得られる．

$$e_{wi} = \begin{cases} \frac{1}{\gamma_w} & \tau^i \text{ が } w \text{ の取りうる品詞の場合} \\ 0 & \text{その他} \end{cases} \quad (9)$$

ここで，γ_w は単語 w が持ちうる品詞の数である☆．

出力 OPT は以下のように定義されるパターンである．

$$OPT = (O_1, O_2, \cdots, O_\gamma) \quad (10)$$

但し，OPT は以下のようにデコードされるものとする．

$$\tau_N(w_t) = \begin{cases} \tau^i & O_i = 1, \text{ かつ すべての } \\ & O_j = 0 \text{（} j \neq i \text{）の場合} \\ Unknown & \text{その他} \end{cases} \quad (11)$$

ここで $\tau_N(w_t)$ は単語 w_t へのタグづけ結果を表す．

ニューロタガーに可変長文脈で品詞タグづけを行なわせるために，ニューロタガーの入力に伸縮性を持たせた．具体的には，はじめに入力 IPT の長さ $l+1+r$ を最大に設定し，タグづけを行なう．タグづけの結果 $\tau_N(w_t)$ が $Unknown$ ならば，入力 IPT の長さ $l+1+r$

☆ 2 節にも述べたように，実験に用いたコーパスに未知語が存在しないため，本論文の議論では未知語の取り扱いを考慮していない．しかし，単語 w が未知語の場合各要素 e_{wi} を $\frac{1}{\gamma}$ に設定すれば未知語の対応も可能と思われる．但し，γ は品詞の数である．

を一定の間隔で縮小してタグづけ処理をもう一度行なう．この処理は $\tau_N(w_t)$ が $Unknown$ でなくなるか，入力の長さ $l+1+r$ が 1（即ち，入力は目標単語のみ）に縮むか，になるまで繰り返される．このような入力の伸縮性の導入は，結果の確信度を高めるために，タグづけをまずできるだけ長い文脈を用いて行ない，訓練データの不足から確定的な答えが出ない場合に順次文脈を短くするといったように長さ優先可変文脈を用いることを意味する．

文の各単語を左から右へ順にタグづけしていくとき，左側の単語はつねにタグづけ済みと考えられるため，それらの単語に関する入力を構成するとき，より多くの情報が活用できる．具体的には，(6)-(9) を用いる代わりに，入力は次のように構成される．

$$ipt_{t-i} = g_{t-i} \cdot OPT(-i) \tag{12}$$

但し，$i = 1, \cdots, l$．ここで，$OPT(-i)$ は i 個前の単語に対してタガーが出した出力を表す．しかしながら，訓練過程においてはタガーの出力はまだ正確ではないため，それらを直接入力にフィードバックして使うことができない．そのために，訓練過程における入力は以下のように実際の出力と目標出力の重みつき平均を用いて構成する．

$$ipt_{t-i} = g_{t-i} \cdot (w_{OPT} \cdot OPT(-i) + w_{DES} \cdot DES) \tag{13}$$

ここで，DES は目標出力で，w_{OPT} と w_{DES} はそれぞれ次のように定義される．

$$w_{OPT} = \frac{E_{OBJ}}{E_{ACT}} \tag{14}$$

$$w_{DES} = 1 - w_{OPT} \tag{15}$$

ここで，E_{OBJ} と E_{ACT} はそれぞれ目標誤差と実際の誤差（詳細は文献 22) を参照）を表す．従って，訓練の始めの入力構成では目標出力の比重が大きく，時間が経つにつれゼロへ減っていく．逆に，実際の出力の比重は最初小さく，時間が経つにつれて大きくなっていく．

3.1.2 インフォメーションゲイン（IG）

インフォメーションゲイン（IG）は，特徴ベクトルで定義されるデータセットの情報量がある特定の特徴の値を知ることによってどれだけ増えるかを表す量である[23),24)]．より具体的に言えば，ある特徴の IG とはその特徴がデータのクラス同定にどれだけ重要かを反映する量である．ここで，特徴を入力の構成部分，特徴の値をその構成部分の取りうる品詞，データの属するクラスを目標単語の取りうる品詞にそれぞれ置き換えてやれば，各構成部分の IG はその構成部分の品詞タグづけへの影響度として考えることができる．

従って，式 (5) における入力の各構成部分 ipt_x（$x = t-l, \cdots, t, \cdots, t+r$）は式 (6) にあるようなそれぞれタグづけへの影響度に応じた重み g_x を持つと仮定すれば，その重みは以下のように求められる．ここで全訓練データのセットを S，i 番目のクラス，即ち，i 番目の品詞（$i = 1, \cdots, \gamma$）を C_i, C_i に属するデータのセットを S_{C_i} で表す．セット S のエントロピー，即ち，S の中の一つのデータのクラス（品詞）を同定するのに必要とされる情報の平均量は

$$info(S) = -\sum_{i=1}^{\gamma} \frac{C(S_{C_i})}{C(S)} \times \ln \frac{C(S_{C_i})}{C(S)} \tag{16}$$

である．但し，$C(X)$ はセット X の中のデータの数を表す．セット S が構成部分 ipt_x の持ちうる h 個の品詞によって h 個のサブセット S_i $(i = 1, \cdots, h)$ に分割されたとき，新しいエントロピーはこれらのサブセットのエントロピーの重みつき総和で求められる．即ち，

$$info_x(S) = \sum_{i=1}^{h} \frac{C(S_i)}{C(S)} \times info(S_i) \tag{17}$$

この分割（即ち，構成部分 ipt_x の品詞を知ること）による情報の増益（IG）は以下になる．

$$gain(x) = info(S) - info_x(S) \tag{18}$$

従って，構成部分 ipt_x のタグづけへの影響度に応じた重みは以下のように設定できる．

$$g_x = gain(x) \tag{19}$$

3.1.3 訓　　練

入力を縮めてもタグづけに同じパラメータ（ユニット間の結合強度: w_{ij}）セットを用いられるようにするために，以下のような訓練方法を取った．ここで，伸縮性入力を持つニューロタガーは，最小入力を持つネットワーク（入力層–中間層–出力層にそれぞれ $N_{Imin} - N_{Hmin} - N_O$ 個のユニットを持つ三層パーセプトロン）から成長してきたもの（入力層–中間層–出力層にそれぞれ $N_{Imax} - N_{Hmax} - N_O$ 個のユニットを持つ三層パーセプトロン）としてとらえる．但し，N_{Imin} と N_{Imax} はそれぞれ最小入力と最大入力を持つネットワークの入力層のユニット数，N_{Hmin} と N_{Hmax} はそれぞれ最小入力と最大入力を持つネットワークの中間層のユニット数，そして N_O はすべてのネットワークの出力層のユニット数である．従って，伸縮性入力ニューロタガーの訓練は小さいパーセプトロンの訓練から最大パーセプトロンの訓練への漸進型過程として考える．具体的には，まず最小入力を持つパーセプトロン（例えば図 3 の実線の部分）を訓練する．訓練し終えた後，それを段階的に成長させ，新し

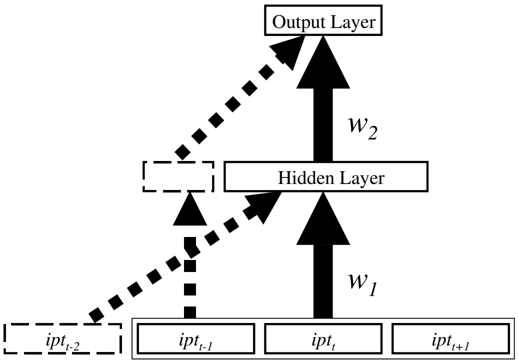

図 3　伸縮性ニューロタガーの訓練
Fig. 3　Training of the elastic neuro tagger

いパーセプトロンを形成する（例えば図 3 の点線を含む全体）．それから，その新しいパーセプトロンの訓練を行なう．そのとき，前の訓練で得たネットワークの結合強度のセット（例えば図 3 の w_1 と w_2）はそのまま初期値として用いられる．言うまでもなく，この方法を用いてもパーセプトロンの性質から，w_1 と w_2 は新しい学習によって多少修正される．しかしながら，最初から最大入力のパーセプトロンを訓練する場合に比べれば，その変動の程度（つまり，最小入力のパーセプトロンの単独訓練で得られるものとの違い）が小さいと思われる．

3.2　ルールベースコレクター

ニューロタガーは，単語の品詞が前の単語のみによって確実に決まる場合でも，あくまでも文脈全体の下での可能性に基づいて解析を行なう．即ち，一般的に言えばニューロタガーがコーパスからの学習で獲得しやすいのが基本的に論理積関係，即ち，$(ipt_{t-l}\ \&\ \cdots\ ipt_t\ \&\ \cdots\ ipt_{t+r} \rightarrow OPT)$，のような規則であり，単項式関係の規則，例えば，$(ipt_x \rightarrow OPT)$，を獲得するのが困難☆である．その結果，前の単語が同じでも全体の文脈が変わると，タグづけ結果が変わってしまう可能性がある．また，語彙情報が品詞タグづけに非常に重要な情報であるにもかかわらず，ニューロタガーはその規模が膨大になることから語彙情報を用いるのが困難である．即ち，ニューロタガーは，$(w \rightarrow OPT)$ や $(w\ \&\ \tau \rightarrow OPT)$，或は $(w_1\ \&\ w_2 \rightarrow OPT)$ のような単語そのものを条件とする規則を獲得することができない．ここで，w,w_1,w_2 は単語を表し，τ は品詞を表している．更に，ニューラルネットはその収束性と過学習問題から訓練データへの品詞タグづけ精度を 100%まで学習し続けるのはほとんど不可能であり得策ではないことから，適当と思われる精度で学習を止めるのが普通である．そのため，論理関係を問わず有用な規則が獲得されていない恐れがある．

ニューロタガーの以上のような弱点を補うために，書き換え規則に基づく修正器（ここでルールベースコレクターと呼ぶ）を後処理として導入する．書き換え規則は，テンプレートセットをまず用意し，その個々のテンプレートを訓練用コーパスへのニューロタガーによるタグづけのエラー箇所に適用することによって，全コーパスを通し，正しく修正された箇所の数と間違って修正された箇所の数の差が最大になるような規則を選び出すことによって獲得される．テンプレートは以上に述べたニューロタガーの欠点を補完するように単項入力の規則，語彙情報を入力に用いる規則，更には品詞と語彙の論理積を入力とした規則から構成される．表 1は本システムに用いたテンプレートセット☆☆を示す．具体的な学習手続きは表 2に示す．

このような統合システムによるタグづけは以下のように行なわれる．タグづけしようとするコーパスが与えられたとき，まずそれをニューロタガーによってタグづけする．そしてタグづけされたコーパスが表 2 に示している学習手続きで獲得した順序付書き換え規則で修正される．その修正は個々の規則を順番にコーパスに適用してはコーパス（の品詞タグ）を更新するといった繰り返し過程である．

4. 実 験 結 果

実験用データは文献 21)，22) と同様，すでに品詞のタグづけが行なわれたタイ語コーパスから得られた 10,452 の文であった．それを無作為に 8,322 文と 2,130 文に分けてそれぞれ訓練とテストに使った．訓練文においては 124,331 個の単語（そのうち，22,311 個の単語が複数の品詞を持つ），テスト文において 34,544 個の単語（そのうち，6,717 個の単語が複数の品詞を持つ）を有する．ニューロタガーの訓練には訓

☆ 3.1 節にも述べたようにニューロタガーの入力は伸縮可能なので，(l,r) を $(0,0)$ へ縮小する場合，ニューロタガーが単独の入力 ipt_t でタグづけを行なうことになる．その意味では，ニューロタガーも単項式関係を扱える．しかしながら，ここで言う単項式関係はより一般的な場合，即ち，単独の入力は任意の ipt_x $(x = t-l, \cdots, t+r)$ である．また，ニューラルネットは理論的に任意の関係が学習可能なので，ここで「困難」という言葉を用いる．

☆☆ このテンプレートセットが適切かどうかを評価するために，単語の論理和入力を加えたテンプレートセットや更に品詞の論理積と論理和入力を加えたテンプレートセット，そして，語彙情報なしのテンプレートセット（表 5を参照）などを用いた追加実験もした．結果は 4 節に述べる．

表 1 書き換え規則のテンプレートセット
Table 1 The set of templates for transformation rules

タグ τ^a をタグ τ^b へ変更する，もし
（単項入力）
（入力は品詞）
1．左（右）の単語のタグが τ である
2．二つ左（右）の単語のタグが τ である
3．三つ左（右）の単語のタグが τ である
（入力は単語）
4．目標単語が w である
5．左（右）の単語が w である
6．二つ左（右）の単語が w である
（単語の論理積入力）
7．目標単語が w_1 で，左（右）の単語が w_2 である
8．左（右）の単語が w_1 で，二つ左（右）の単語が w_2 である
9．左の単語が w_1 で，右の単語が w_2 である
（品詞と単語の論理積入力）
10．目標単語が w で，左（右）の単語のタグが τ である
11．左（右）の単語が w で，左（右）の単語のタグが τ である
12．目標単語が w_1 で，左（右）の単語が w_2 で，左（右）の単語のタグが τ である

表 2 書き換え規則の学習手続き
Table 2 Learning procedure of transformation rules

1．ニューロタガーで訓練用コーパスをタグづけし，コーパスを更新する
2．タグづけた結果と正解を比較し，エラー箇所を見つける
3．個々のエラー箇所において，テンプレートとの照合を行ない，規則群を得る
4．最適な規則を $(cnt\_good - h \cdot cnt\_bad)$ が最大であるように選ぶ．但し， cnt_good: 間違ったタグを正しい方へ変更する数 cnt_bad: 正しいタグを間違った方へ変更する数 h: 規則を生成する厳格さを制御する重み
5．最適な規則を訓練コーパスへ適用し，コーパスを更新する
6．最適な規則を順序付書き換え規則のリストに付け加える
7．最適な規則がなくなる（$cnt\_good - h \cdot cnt\_bad \leq 0$）まで手順 2 - 6 を繰り返す

練文の中の品詞の曖昧性のある単語のみを用いた．HMM の訓練には訓練文の全単語を用いた．但し，いずれの場合においても各単語がそれぞれの品詞を取る頻度 $Prob(\tau^i|w)$ の推定には訓練文の全単語を用いた（HMM の詳細については文献 22) を参照）．タイ語には 47 種類の品詞が定義されている [26] ため，γ [式 (2)] は 47 にセットされる．

ニューロタガー 入力層–中間層–出力層に $p - \frac{p}{2} - \gamma$ 個のユニットを持つ三層パーセプトロンで構成される．ここで，$\gamma = 47$，$p = \gamma \cdot (l+1+r)$．但し，$(l+1+r)$ は以下のように伸縮性をもつ．訓練段階においては，(l,r) を (1,1) $\rightarrow$ (2,1) $\rightarrow$ (2,2) $\rightarrow$ (3,2) $\rightarrow$ (3,3) のように段階的に増加させ，小さいネットワークから大きいネットワークへ漸進的に訓練を行なう．一方，タグづけにおいては，逆に必要に応じて (l,r) を (3,3) $\rightarrow$ (3,2) $\rightarrow$ (2,2) $\rightarrow$ (2,1) $\rightarrow$ (1,1) $\rightarrow$ (1,0) $\rightarrow$ (0,0) のように段階的に縮小していく．但し，タグづけにおいては，中間層のユニット数を最大のまま（即ち，$(l,r) = (3,3)$ に対応したもの）に固定した☆．

ルールベースコレクター 表 2 の学習手続きに用いた規則生成の評価関数 $cnt\_good - h \cdot cnt\_bad$ のパラメター h は規則生成の厳しさを制御するものである☆☆．パラメター h を大きく設定すると，cnt_bad の影響が大きくなり，少しでも間違いを生じるような規則は生成されにくくなる．本稿では，ニューロタガーはすでに高い精度を持ち，ルールベースコレクターはあくまでも微調整のチューナという位置づけで用いる（即ち，規則の生成を厳しくする）ため，重み h を 100 という大きな値に設定した．訓練データを，表 1に示すテンプレートを用いて表 2に示す手続きで学習した結果，計 520 個の順序つき書き換え規則が得られた．表 3はその最初の 15 個の規則を示す．

表 4はテストデータへのタグづけ結果を示している．表に示す精度は品詞に曖昧性のある単語のみを対象に測定されたものである．表には伸縮性ニューロタガーと統合システムの精度を示しているほか，これらとの比較のため，ベースラインモデル，HMM，そしてルールベースモデルのそれぞれの精度も示している．ここで言うベースラインモデルとは，文脈情報を使わず訓練コーパスから得られた個々の単語が取る品詞の頻度情報のみを用いてタグづけを行なうものである．また，ルールベースモデルとは，ニューロタガーを用いる代わりにベースラインモデルのタグづけ結果に，表 5に示すテンプレートを用いて表 2 に示す手続きで訓練データを学習して得られた 1,177 個の書き換え規則を適用したものである．但し，この場合，書き換え規則は微調整チューナではないため，h を 1 に設定した．

表に示しているように，伸縮性ニューロタガーの精度は 94.4%で，HMM のそれより遥かに高く，ルールベースモデルのそれよりも高かった．一方，ルールベースモデルの精度はニューロタガーのそれに劣るものの，統計手法のそれより遥かに高かった．そして，統合システムの精度は 95.5%で，伸縮性ニューロタガーのそれより 1.1%高かった．実際，書き換え規則は，訓練データとテストデータへのニューロタガーによるタグ

☆ 実際，中間層のユニット数を入力の長さに応じて変化させる方法を用いてもほぼ同じ実験結果が得られた．
☆☆ Brill の論文 5) では h のようなパラメターを用いず $cnt\_good - cnt\_bad$ を評価関数とした．

表 3 最初の 15 個の書き換え規則
Table 3 The first fifteen transformation rules

No.	From	To	Condition
1	PREL	RPRE	左の単語が句読点で、右の単語がระดับである
2	PREL	RPRE	左の単語がอยู่である
3	Unknown	ADVN	左の単語のタグがXVAEである
4	XVMM	XVBM	左の単語がหรือである
5	VATT	ADVN	左の単語がกันである
6	Unknown	VATT	左の単語のタグがPRELである
7	NCMN	RPRE	左の単語がผลである
8	VATT	VSTA	左の単語がสามัญである
9	PREL	RPRE	右の単語がระดับで、二つ右の単語がความである
10	VSTA	ADVN	目標単語がต่อเนื่องである
11	VATT	ADVN	目標単語がสูงสุดである
12	NCMN	RPRE	目標単語がทางで、左の単語がออกแบบである
13	NCMN	RPRE	左の単語がในで、左の単語のタグがNCMNである
14	Unknown	ADVN	三つ左の単語のタグがVACTである
15	NCMN	CNIT	目標単語がภาพである

但し、PREL: Relative Pronoun, RPRE: Preposition, ...

表 4 テストデータへの品詞タグづけ結果 *
Table 4 The results of POS tagging

モデル	ベースライン	HMM	ルールベース	伸縮性ニューロ	統合システム
精度	0.836	0.891	0.935	0.944	0.955

∗ 品詞に曖昧性のある単語のみを測定対象とした．

づけのエラーの 88.4%と 19.7%をそれぞれ修正した．また，伸縮性ニューロタガーの精度はどの固定長入力のニューロタガーのそれ[詳細は 22) を参照] よりも高かった．従って，伸縮性ニューロタガーは，文脈の長さを事前に経験的に選ぶ必要がなく，いつも状況に応じて適切な長さの文脈を自動的に選べる．

表 1のテンプレートセットは主にニューロタガーの弱点（単項式関係の規則の獲得と語彙情報の使用が困難である）を補うために設計されたもので，Brill が用いたテンプレートセット 5) に比べテンプレートの数はかなり少ない．このような小規模なテンプレートセットが十分かどうか，または適切かどうかを検証するために，表 1のテンプレートセットと Brill のテンプレートセットの和集合のテンプレートセット（表 5）を用意してそこから作成した幾つかのサブセットを用いた比較実験を行なった．そのとき，表 1のテンプレートセットは表 5のテンプレートセットのテンプレート 1–9 とテンプレート 13–15 から構成されているサブセットとして考えることができる．

まず，表 1のテンプレートセットが十分かどうかを見るために，そのテンプレートセットの上に（1）単語の論理和入力のテンプレート（表 5の 10–12）を追加したテンプレートセットと（2）品詞の論理和入力と論理積入力のテンプレートも加えたテンプレートセット（表 5のテンプレートセットそのもの）を用いた追加実験を行なった．実験の結果，（1）の場合のタグづけ精度は全く変わらなかった；そして（2）の場合のタグづけ精度は僅か 0.03%しか向上しなかった（即ち精度表示は同じ 95.5%であった）．以上の結果から，表 1のテンプレートセットはほぼ十分であることが分かった．それから，表 1のテンプレートセットが適切かどうかを見るために単語の論理積入力のテンプレート（表 1の 7–9）の代わりに単語の論理和入力のテンプレート（表 5の 10–12）を用いたテンプレートセットを用いた追加実験を行なった．その結果，タグづけ精度が 95.4%で 0.1%落ちた．従って，単語の論理和入力のテンプレートより論理積のテンプレートを用いたほうが適切であることが分かった．一方，語彙情報のないテンプレートセット（即ち，表 5のテンプレートセットから語彙情報が入っているテンプレート 4–15 を取り除いたテンプレートセット）を用いた追加実験も行なった．その結果，タグづけ精度は 0.9%落ち，94.6%であった．従って，品詞タグづけに語彙情報を用いるのが重要であることが分かった．更に，規

表 5 書き換え規則のテンプレートセット（比較実験用）
Table 5 The set of templates for transformation rules used in comparative experiments

タグ τ^a をタグ τ^b へ変更する，もし
（単項入力）
（入力は品詞）
１．左（右）の単語のタグが τ である
２．二つ左（右）の単語のタグが τ である
３．三つ左（右）の単語のタグが τ である
（入力は単語）
４．目標単語が w である
５．左（右）の単語が w である
６．二つ左（右）の単語が w である
（単語の論理積入力）
７．目標単語が w_1 で，左（右）の単語が w_2 である
８．左（右）の単語が w_1 で，二つ左（右）の単語が w_2 である
９．左の単語が w_1 で，右の単語が w_2 である
（単語の論理和入力）
１０．目標単語と左（右）の単語のどれかが w である
１１．左（右）と二つ左（右）の単語のどれかが w である
１２．左と右の単語のどれかが w である
（品詞と単語の論理積入力）
１３．目標単語が w で，左（右）の単語のタグが τ である
１４．左（右）の単語が w で，左（右）の単語のタグが τ である
１５．目標単語が w_1 で，左（右）の単語が w_2 で，左（右）の単語のタグが τ である
（品詞の論理和入力）
１６．左と右の単語のどれかのタグが τ である
１７．左（右）と二つ左（右）の単語のどれかのタグが τ である
１８．左（右）と二つ左（右）と三つ左（右）の単語のどれかのタグが τ である
（品詞の論理積入力）
１９．左の単語のタグが τ_1 で，右の単語のタグが τ_2 である
２０．左（右）の単語のタグが τ_1 で，二つ左（右）の単語のタグが τ_2 である

則を生成する厳しさを制御する重み h を大きくしたのが正しいかどうかを確認するために，表 1のテンプレートセットを用いて $h=1$ の追加実験を行なった．その場合，$h=100$ の場合に比べタグづけ精度は確かに微小に（0.045%）低下したが，有意差が認められるほどのものではなかった．

最後に注意してほしいのは，これまでの文献では品詞タグづけの精度としてよくテストデータの全単語（品詞の曖昧性のありなし問わず）を対象に測定して得たものを用いているということである．もしこのような定義を用いれば，統合システムの精度は99.1%に達することになる．

5. 結 び

伸縮性ニューロタガーとルールベースコレクターで構成される統合品詞タグづけシステムを提案した．伸縮性ニューロタガーは情報量最大を考慮し動的に適切な長さの文脈でタグづけができる．一方，ルールベースコレクターはニューロタガーが獲得困難な規則を誤り駆動型学習で自動獲得し，ニューロタガーが生じたエラーを修正する．計算機実験の結果，伸縮性ニューロタガーは HMM やルールベースモデルより高い精度（94.4%）でタグづけができた．そして，書き換え規則を後処理に導入したことによりタグづけのエラーは 19.7%減少し，タグづけ精度は 95.5%まで向上した．この精度は多品詞語のみを測定対象として得られたもので，全単語を対象に測定した場合，タグづけ精度は 99.1%になる．従って，数万オーダーの多品詞語を含む小規模コーパスを訓練に用いても，本システムのタグづけ精度はほぼ実用的に使えるレベルに達していると言える．

参 考 文 献

1) 内元，馬：形態素構文解析，人文学と情報処理，1999．
2) 村田，井佐原：意味文脈解析，人文学と情報処理，1999．
3) R. Garside, G. Leech, and G. Sampson: *The computational analysis of English: A corpus-based approach*, London: Longman (1987).
4) D. Hindle: "Acquiring disambiguation rules from text." *Proc. ACL'89*, Vancouver BC, pp. 118-125 (1989).
5) Brill, E.: Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging, *Computational Linguistics*, Vol. 21, No. 4, pp. 543-565, 1994.
6) K. Church: "A stochastic parts program and noun phrase parser for unrestricted text." *Proc. 2nd ACL Applied NLP*, Austin, Texas, pp. 136-143 (1988).
7) S. DeRose: "Grammatical category disambiguation by statistical optimization." *Computational Linguistics*, Vol. 14, No. 1, pp. 31-39 (1988).
8) D. Cutting, J. Kupiec, J. Pederson, and P. Sibun: "A practical part of speech tagger." *Proc. 3rd ACL Applied NLP*, Trento, Italy, pp. 133-140 (1992).
9) E. Charnik, c. Hendrickson, N. Jacobson, and M. Perkowitz: "Equations for part-of-speech tagging." In *Proc. 11th National Conference on Artificial Intelligence*. AAAI Press/MIT Press, Menlo Park, pp. 784-789 (1993).
10) E. Charniak: *Statistical language learning*, The MIT Press (1993).

11) R. Weischedel, et al.: "Coping with ambiguity and unknown words through probabilistic models." *Computational Linguistics*, Vol. 19, No. 2, pp. 359-382 (1993).
12) B. Merialdo: "Tagging English text with a probabilistic model." *Computational Linguistics*, Vol. 20, No. 2, pp.155-171 (1994).
13) H. Schütze and Y. Singer: "Part-of-speech tagging using a variable memory markov model." *Proc. ACL'94*, Las Cruces, New Mexico, pp. 181-187 (1994).
14) M. Nakamura, K. Maruyama, T. Kawabata, and K. Shikano: "Neural network approach to word category prediction for English texts." *Proc. COLING'90*, Helsinki University, pp. 213-218 (1990).
15) H. Schmid: "Part-of-speech tagging with neural networks." *Proc. COLING'94*, Japan, pp. 172-176 (1994).
16) Q. Ma, H. Isahara, and H. Ozaku: "Automatic part-of-speech tagging of Thai corpus using neural networks", *Artificial Neural Networks – ICANN'96*, Lecture Notes in Computer Science 1112, C. von der Malsburg et al (Eds.), Springer, pp. 275-280 (1996).
17) A. Ratnaparkhi: " A maximum entropy model for part-of-speech tagging." In *Proc. Conf. on Emprical Methods in Natural Language Processing.* University of pennsylvania, pp. 133-142 (1996).
18) W. Daelemans, J. Zavrel, P. Berck, and S. Gillis: "MBT: A memory-based part of speech tagger-generator." *Proc. 4th Workshop on Very Large Corpora*, Copenhagen, Denmark, pp. 1-14 (1996).
19) L. Marquez and L. Padro: "A flexible POS tagger using an automatically acquired language model." *Proc. ACL-EACL'97*, Madrid, Spain, pp. 238-252 (1997).
20) Q. Ma and H. Isahara: "A multi-neuro tagger using variable lengths of contexts", *Proc. COLING-ACL'98*, Montreal, pp. 802-806 (1998).
21) 馬, 井佐原: "長さ可変文脈を用いたマルチニューロタガー", 自然言語処理, vol. 6, No. 1. pp. 29-42, 1999.
22) 馬，内元，村田，井佐原：品詞自動タグづけシステム–伸縮性入力ニューロタガー，人工知能学会誌，Vol. 14, No. 6, pp. 1116-1124 (1999)．
23) W. Daelemans and A. Van den Bosch: "Generalisation performance of backpropagation learning on a syllabification task." In M. Drossaers & A. Nijholt (Eds.), *TWLT3: Connectionism and Natural Language Processing.* Enschede: Twente University, pp. 27-38 (1992).
24) J. Quinlan: *C4.5: Programs for Machine Learning*, San Mateo, CA: Morgan Kaufmann (1993).
25) 久光，丹羽： 書き換え規則と文脈情報を用いた形態素解析後処理，NL 研 126-8，pp. 55-62，1998-7．
26) T. Charoenporn, V. Sornlertlamvanich, and H. Isahara: "Building a large Thai text corpus - part of speech tagged corpus: ORCHID." *Proc. Natural Language Processing Pacific Rim Symposium 1997*, Phuket, Thailand, pp. 509-512 (1997).

(平成？年？月？日受付)
(平成？年？月？日採録)

馬　青

1983 年北京航空航天大学自動制御学部卒業．1987 年筑波大学大学院理工学研究科修士課程修了．1990 年同大学院工学研究科博士課程修了．工学博士．1990 ～ 93 年株式会社小野測器勤務．1993 年郵政省通信総合研究所入所，主任研究官．人工神経回路網モデル，知識表現，自然言語処理の研究に従事．日本神経回路学会，自然言語処理学会，電子情報通信学会，各会員．

内元　清貴（正会員）

1994 年京都大学工学部卒業．1996 年同大学院修士課程修了．同年郵政省通信総合研究所入所．研究官．自然言語処理の研究に従事．言語処理学会，情報処理学会，ACL，各会員．

村田　真樹（正会員）

1993 年京都大学工学部卒業．1995 年同大学院工学研究科修士課程修了．1997 年同大学院工学研究科博士課程修了．工学博士．同年京都大学にて日本学術振興会リサーチ・アソシエイト．1998 年郵政省通信総合研究所入所．自然言語処理，情報検索，機械翻訳の研究に従事．情報処理学会，人工知能学会，言語処理学会，ACL，各会員．

井佐原　均（正会員）

1978年京都大学工学部卒業．1980年同大学院工学研究科修士課程修了．工学博士．同年通商産業省電子技術総合研究所入所．1995年郵政省通信総合研究所入所．2000年同けいはんな情報通信融合研究センター勤務．自然言語処理，機械翻訳の研究に従事．情報処理学会，言語処理学会，人工知能学会，日本認知科学会，ACL，各会員．