情報量最大を考慮し最長文脈を優先するタイ語ニューロタガー

馬 青　井佐原 均
郵政省通信総合研究所
{qma, isahara}@crl.go.jp

要旨

情報量最大を考慮し最長文脈優先に基づいて長さ可変文脈で品詞タグづけを行うマルチニューロタガーを提案する．マルチニューロタガーは，10,452文の小規模タイ語コーパスを訓練に用いることにより，未訓練タイ語データを94%以上の正解率でタグづけすることができる。

1 はじめに

自動品詞タグづけは自然言語のテキスト処理に限らず音声処理や情報検索など幅広い分野において必要不可欠な要素技術である．膨大な訓練データ（例えば英語の場合は1,000,000オーダー）を用いれば，これまで提案された品詞タグづけシステムはすでに95%程度の正解率を得ている．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．従って，これらの言語にとっては，如何に少ない訓練データで十分実用的で高い正解率の品詞タグづけシステムを構築するかが重要な課題となる．

これまで提案された確率モデルやニューラルネットモデルのほとんど（例えば，Merialdo, 1994; Schmid, 1994）はタグづけに長さが固定の文脈を用いるものであり，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．しかし，訓練データが少ない場合には，タグづけ結果の確信度を高めるために，まずできるだけ長い文脈を用い，そこで確定的な答えが出ない場合に文脈を短くするといったようにフレキシブルにタグづけすることが必要とされよう．そして，客観的な基準で入力の各構成部分の品詞タグづけへの影響度を計り，その影響度に応じた重みを入力のそれぞれの構成部分に与えることができればより望ましいと思われる．本稿では，複数のニューラルネットで構成されるマルチニューロタガーを提案する．品詞のタグづけは，長さが固定の文脈を用いる代わりに，最長文脈優先でフレキシブルに行なわれる。個々のニューラルネットの訓練はそれぞれ独立に行なわれるのではなく，短い文脈での訓練結果（訓練で獲得した重み）を長い文脈での初期値として使う。その結果，訓練時間が大幅に短縮でき，複数のニューラルネットを用いても訓練時間上問題が生じない。ある単語の品詞を決定する場合，その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を与えている。従って，情報量最大を考慮して訓練データからインフォメーションゲイン（略してＩＧと呼ぶ）を求め，それを影響度として各入力の構成部分を重み付けする。その結果，訓練時間が更に短縮され，タグづけの性能も改善される。

2 マルチニューロタガー

2.1 品詞タグづけ問題

一つ一つの単語はしばしば複数の品詞（即ち，品詞の曖昧性）を持ち得る．しかしながら，その単語が一旦文に組み込まれば，持ち得る品詞はその前後の品詞によって唯一に決まる場合が多い。入力されるタイ語テキストは電子辞書によって単語に分割され，各単語の持ちうる品詞がリストアップされる．ここで品詞タグづけ問題は以下に示すような曖昧性除去或は一種のクラス分け問題と見なせる：

$$IPT : (ipt\_l_l, \cdots, ipt\_l_1, ipt\_t, ipt\_r_1, \cdots, ipt\_r_r)$$

$$\Rightarrow OPT : POS\_t. \qquad (1)$$

ここで，ipt_t は目標単語の取りうる品詞に関する入力部分，$(ipt\_l_l, \cdots, ipt\_l_1)$ と $(ipt\_r_1, \cdots, ipt\_r_r)$ はそれぞれ左と右にある単語の取りうる品詞（文脈）に関する入力部分，そして，POS_t は目標単語がその文脈で取りうる正しい品詞を表す．

2.2 インフォメーションゲイン（IG）

入力の各構成部分 ipt_x ($x = l_i, t,$ or r_j) 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 (IG: Daelemans & van de Bosch, 1992; Quinkan, 1993) で重み付けられる．ここで訓練データセットを S, i 番目のクラス（品詞）を C_i で表す ($i = 1, \cdots, n$, n は品詞の数)．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

$$info(S) = -\sum_{i=1}^{n} \frac{freq(C_i, S)}{|S|} \times ln(\frac{freq(C_i, S)}{|S|}) \quad (2)$$

である．但し，$|S|$ は S の中のデータの数，$freq(C_i, S)$ はそのうちクラス C_i に属するデータの数である．入力部分 ipt_x の持ちうる各品詞によって S が 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以下のようになる：

$$info_x(S) = \sum_{i=1}^{h} \frac{|S_i|}{|S|} \times info(S_i). \quad (3)$$

この分割（即ち，入力部分 ipt_x の品詞を知ること）による情報の増益は以下になる:

$$gain(x) = info(S) - info_x(S). \quad (4)$$

従って，入力部分 ipt_x の重みは以下のように設定される:

$$w\_x = gain(x). \quad (5)$$

2.3 シングルニューロタガー（SNT）

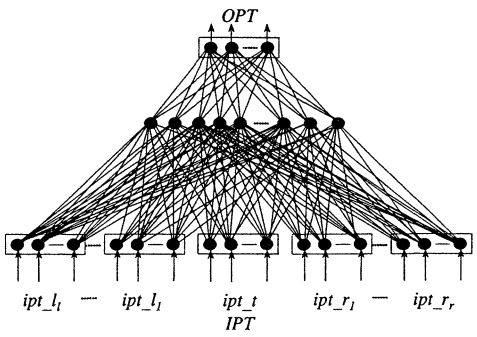

図1 シングルニューロタガー (SNT)

図1は固定長さの文脈を用いて品詞タグづけをするニューラルネット（シングルニューロタガー，略してSNTと呼ぶ）を示す．入力の各構成部分 ipt_x ($x = t, l_i,$ or r_j) は以下のように重み付けされたパターンで定義される:

$$ipt\_x = w\_x \cdot (e_{x1}, e_{x2}, \cdots, e_{xn}) = (I_{x1}, I_{x2}, \cdots, I_{xn}). \quad (6)$$

但し，n はタイ語で定義されている品詞の数であり，$I_{xi} = w\_x \cdot e_{xi}$ ($i = 1, \cdots, n$) である．もし単語 x が既知のもの，即ち，訓練データに出現するならば，各ビット e_{xi} は以下のように得られる：

$$e_{xi} = Prob(POS_i|x). \quad (7)$$

ここで $Prob(POS_i|x)$ は単語 x の品詞が POS_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:

$$Prob(POS_i|x) = \frac{|POS_i, x|}{|x|}. \quad (8)$$

ここで，$|POS_i|$ は全訓練データを通じ，POS_i と x の両方が同時に出現する回数で，$|x|$ は x が出現する回数である．一方，もし単語 x が未知のもの，即ち，訓練データに出現しないならば，各ビット e_{xi} は以下のように得られる：

$$e_{xi} = \begin{cases} \frac{1}{n_x} & POS_i \text{ が } x \text{ の取りうる品詞の場合} \\ 0. & \text{その他} \end{cases} \quad (9)$$

ここで n_x は単語 x 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：

$$OPT = (O_1, O_2, \cdots, O_n). \quad (10)$$

OPT はデコードされ，目標単語の品詞として最終結果 RST が得られる：

$$RST = \begin{cases} POS_i & \text{if } O_i = 1 \ \& \ O_j = 0 \text{ for } j \neq i \\ Unknown. & \text{otherwise} \end{cases} \quad (11)$$

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．従って，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，入力は次ぎのように構成される：

$$ipt\_l_i(t) = w\_l_i \cdot OPT(t - i). \quad (12)$$

ここで，t は目標単語の文における位置であり，$i = 1, 2, \cdots, l$ for $t - i > 0$．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，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づき平均を用いて構成する：

$$ipt\_l_i(t) = w\_l_i \cdot (w_{OPT} \cdot OPT(t - i) + w_{DES} \cdot DES). \quad (13)$$

ここで，DES は目標出力で，w_{OPT} と w_{DES} はそれぞれ次ぎのように定義される重みである：

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

and

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

ここで，E_{OBJ} と E_{ACT} はそれぞれ目標エラーと実際のエラーである．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，時間が立つにつれて大きくなっていく．

2.4 マルチニューロタガー

図2に示すように，マルチニューロタガーはエンコーダー/デコーダー，複数のシングルニューロタガー SNT_i $(i = 1, \cdots, m)$，そして最大文脈優先セレクターで構成される．SNT_i は入力 IPT_i を持つ．入力 IPT_i の長さ（入力部分の数）$l(IPT_i)$ は次ぎの関係を持つ：$l(IPT_i) < l(IPT_j)$ for $i < j$.

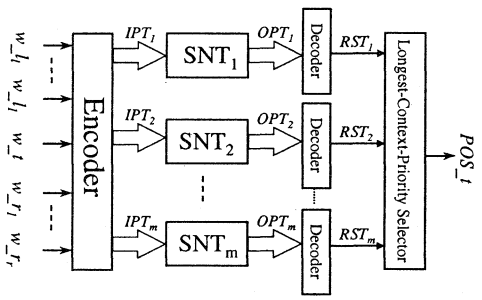

図2 マルチニューロタガー

目標単語 w_t を中心とした，最大長さ $l(IPT_m)$ の単語列 $(w\_l_l, \cdots, w\_l_1, w\_t, \cdots, w\_r_r)$ がマルチニューロタガーに入力される時，それぞれ同じく単語 w_t を中心とした長さ $l(IPT_i)$ の部分単語列が前節に述べた方法で IPT_i に符号化され，個々のシングルニューロタガー $\mathrm{SNT}_i(i = 1, \cdots, m)$ に入力される．個々のシングルニューロタガーの出力 $OPT_i(i = 1, \cdots, m)$ は前節に述べた方法で RST_i に符号化される．その $RST_i(i = 1, \cdots, m)$ は更に最大文脈優先セレクターに入力され，最終結果は次ぎのように得られる：

$$POS\_t = \begin{cases} RST_i, & RST_i \text{ がアンノンでなく，} \\ & \text{すべての } RST_j(j > i) \text{ が} \\ & \text{アンノンである場合} \\ Unknown. & \text{その他} \end{cases} \tag{16}$$

この式は，タグづけの最終結果はできるだけ長い文脈で得られた出力を優先的に用いることを意味する．

2.5 訓練

訓練時間を大幅に短縮させるために，短い入力の SNT_i の訓練結果（訓練で獲得した重み）を長い入力の SNT_{i+1} $(i = 1, \cdots, m-1)$ にコピーして初期値として使う．例えば，SNT_1 がすでに訓練されたとすれば，図3に示すように，その重み w_1 と w_2 を SNT_2 の対応するところ（実線で示した右の部分）にコピーして初期値として使う．

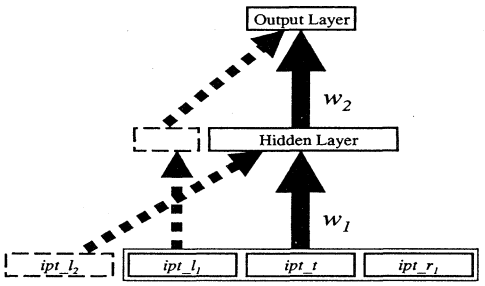

図3 シングルニューロタガー SNT_2 の訓練

2.6 特徴

例えば品詞が50種類ある言語を左右それぞれ三つの単語の情報を文脈としてタグづけを行なう場合，n-gram ベースのモデルは $50^7 = 7.8e + 11$ 個の n-gram（パラメータ）を推定しなければならない．それに対し，ニューロモデル（例えば中間層のユニット数が入力層の半分であるような3層パーセプトロン）は僅か70,000個のパラメータ（重み）しか必要としない．一般的に，必要とされるパラメータの数が少なければ，それらを正しく同定するのに必要な訓練データの数も少ない．そのために，ニューロモデルのタグづけ性能は統計モデルのそれに比べ訓練データの数の少なさに影響されにくいと考えられる（Schmid, 1994）．

表1 テストデータへの品詞タグづけ結果

	シングルニューロタガー					マルチニューロタガー
$l(IPT_i)$	3	4	5	6	7	
IG あり	0.915	0.920	0.929	0.930	0.933	0.943
IG なし	0.924	0.927	0.922	0.926	0.926	0.941

3 実験結果

実験用データは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22,311個の単語が複数の品詞を持ち,テスト文においては6,717個の単語が複数の品詞を持ちえた。タイ語には47種類の品詞 (Charoenporn et al., 1997) が定義されているため,SNT_i は入力層-中間層-出力層に $(l(IPT_i) \times 47) - (l(IPT_i) \times 47/2) - 47$ 個のユニットを持つ3層パーセプトロンであった。マルチニューロタガーは入力の長さ $l(IPT_i) = 2 + i$ の五つのシングルニューロタガー SNT_i $(i = 1, \cdots, 5)$ から構成される。訓練セットから得られた各入力部分の重みは $(w\_l_3, w\_l_2, w\_l_1, w\_t, w\_r_1, w\_r_2, w\_r_3) = (0.575, 0.524, 0.749, 2.667, 0.801, 0.575, 0.649)$ であった.

表1はテストデータへの品詞タグづけ結果を示す. この表から, マルチニューロタガーはIGの有無とは関係なく, どのシングルニューロタガーのそれよりも高い正解率を持つことが分かった. 従って, マルチニューロタガーを用いることによって, 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. また, この表から, IGを用いることによって長い文脈 (入力の長さが5以上) でのタグづけの正解率が上がったことも分かった.

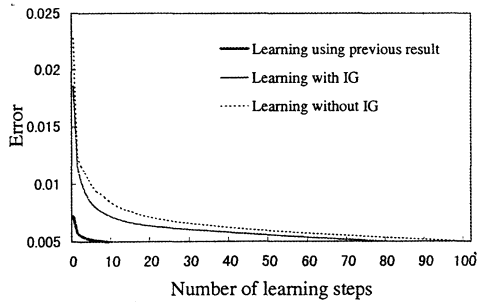

図4 SNT_3 の異なる条件での訓練曲線

図4は異なる条件でのシングルニューロタガー SNT_3 の訓練曲線を示す. 太い実線, 細い実線, そして点線はそれぞれ SNT_2 の訓練結果を利用した場合, SNT_2 の訓練結果を利用しない場合, そして SNT_2 の訓練結果を利用せず, IGも用いない場合である. この図は, 訓練時間の大幅な短縮には前の訓練結果の利用だけでなくIGの利用も効果的であることを示した.

4 結び

情報量最大を考慮し最長文脈優先に基づいて長さ可変文脈で品詞タグづけを行うマルチニューロタガーを提案した. マルチニューロタガーは,10,452文の小規模タイ語コーパスを訓練に用いることにより, 未訓練タイ語データを94%以上の正解率でタグづけすることができた。この結果は, どのシングルニューロタガーを用いた場合よりも優れ, マルチニューロタガーはタグづけ過程において動的に適切な長さの文脈を見つけていることを示した.

参考文献

[1] Merialdo, B.: Tagging English text with a probabilistic model, *Computational Linguistics*, vol. 20, No. 2, pp.155-171, 1994.

[2] Schmid, H.: Part-of-speech tagging with neural networks, *Proc. of the Int. Conf. on Computational Linguistics*, pp. 172-176, 1994.

[3] Daelemans, W. and Van den Bosch, A.: 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.

[4] Quinlan, J.: *C4.5: Programs for Machine Learning*, Morgan Kaufmann, 1993.

[5] Charoenporn, T., Sornlertlamvanich, V., and Isahara, H.: Building a large Thai text corpus - part of speech tagged corpus: ORCHID, Proc. Natural Language Processing Pacific Rim Symposium 1997, Thailand, 1997.