カテゴリ間の遷移情報を用いた文認識系の能力評価法*

7C−5

大槻恭士[1]　伊藤彰則[2]　牧野正三[1]　曽根敏夫[3]
[1]東北大応情研　[2]東北大情教セ　[3]東北大通研

1　まえがき

言語情報を用いて単語ラティスから文を決定する文認識系における単語認識率と文認識率の関係は，タスク中の距離1の文の数より推定することができる(1)が，品詞等のカテゴリのbigramやtrigramの有無を用いた場合の，距離1の文の数の求め方は明らかになっていない．本稿ではそれを求めるアルゴリズムを提案し，単語認識率と文認識率の関係を推定する．

2　認識系のモデル

図1にカテゴリのbigram,trigramを利用した文認識系のモデルを示す．カテゴリのbigram,trigramより生成可能なカテゴリ系列と単語辞書より可能な単語系列を生成し(図2)，単語認識系の出力と照合して文を決定する．

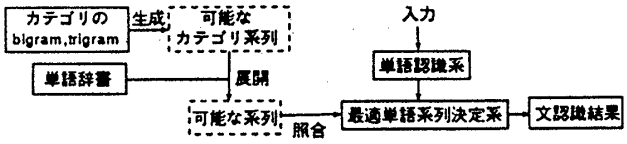

図1: カテゴリのbigram,trigramを利用した文認識系のモデル

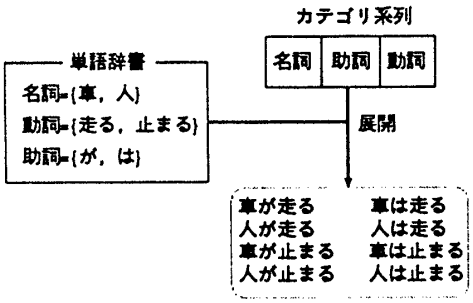

図2: カテゴリ系列の単語辞書による展開

この場合，単語認識率と文認識率の関係は，全単語系列中の距離1の文の数を計算して，我々が提案した評価式(1)によって推定することが可能であるが，その単語系列の数は語彙数とともに大きくなり，単純な距離計算は困難となる．文$S=s_1s_2\cdots s_L$と文$T=t_1t_2\cdots t_L$間の距離$D(S,T)$は，

$$D(S,T)=\sum_{i=1}^{L} f(s_i,t_i)\ ,\quad f(a,b)=\begin{cases}0 & \text{if} \quad a=b\\ 1 & \text{else}\end{cases} \tag{1}$$

と定義される．そこで，可能な全ての単語系列中の距離1の文の数を計算するために，次に述べるようなアルゴリズムを提案する．

3　アルゴリズム

trigramはbigramに変形可能であるから，ここではbigramについて説明を行う．まず，以下のように記号を定義する．

- M: カテゴリの種類数
- $c_1,c_2,\ldots,c_M$: カテゴリ
- c_s,c_e: 始端，終端を表す仮想的なカテゴリ
- L: 文の長さ
- $X=x_1x_2\cdots x_L$: 長さLの文
- x_i: 文Xの始端からi番目の単語
- $N_d(X)$: 文Xと距離がdである単語系列の数
- $\Lambda(c)$: カテゴリcに属する単語の数
- $V(x)$: 単語xの属すカテゴリ
- Ω_l: 長さlの生成可能な全系列
- Ω_l^c:Ω_l中の終端のカテゴリがcのもの
- X_l：Xの部分列$x_1x_2\cdots x_l$
- $N_d(X_l,\Omega_l^c)$:Ω_l^c中のX_lとの距離がdである系列の数
- s_c:cを出力する状態
- $I_t(c,c')$: 遷移情報

$$I_t(c,c')=\begin{cases}1 & \text{if } s_c\rightarrow s_{c'} \text{ is possible}\\ 0 & \text{if } s_c\rightarrow s_{c'} \text{ is impossible}\end{cases}$$

文Xも当然生成可能な単語系列であるから$X_l\in\Omega_l$であり，明らかに次式が成り立つ．

$$N_0(X_l,\Omega_l^c)=\begin{cases}1 & \text{if} \quad c=V(x_l)\\ 0 & \text{else}\end{cases} \tag{2}$$

ここでΩ_{l-1}中の1つの系列を考える．これが次にs_cに遷移したとき$\Lambda(c)$個の系列に展開されるが，ここで$c=V(x_l)$なら，展開された系列の中の終端がx_lの系列1つはX_lとの距離は遷移前のX_{l-1}との距離と同じである．しかし，残りの$\Lambda(c)-1$個の系列では，X_lとの距離は遷移前のX_{l-1}との距離より1増加したものになる．よって，$N_1(X_l,\Omega_l^c)$はΩ_{l-1}のうちX_{l-1}と距離1でs_cに遷移が可能な系列の数と，Ω_{l-1}のうちX_{l-1}と距離0でs_cに遷移が可能な系列の数に$\Lambda(c)-1$を掛けたものとの和とな

*The performance evaluation method on sentence recognition system which uses the transition information between word categories. by T.Otsuki[1], A.Ito[2], S.Makino[1] and T.Sone[3]. [1]R.C.A.I.S., Tohoku Univ. [2]E.C.I.P., Tohoku Univ. [3]R.I.E.C., Tohoku Univ.

る．一方，$c \neq V(x_l)$ なら，展開された全ての系列で，X_l との距離が X_{l-1} との距離より1増加したものになる．よって，$N_1(X_l, \Omega_l^c)$ は Ω_{l-1} のうち X_{l-1} と距離0で s_c に遷移が可能な系列の数に $\Lambda(c)$ を掛けたものとなる．以上を式で表すと次式となり，$N_1(X_l, \Omega_l^c)$ は逐次的に計算が可能である．

$$N_1(X_l, \Omega_l^c) = \begin{cases} \sum_{i=1}^{M} \{N_1(X_{l-1}, \Omega_{l-1}^{c_i}) \\ \quad +(\Lambda(c)-1)N_0(X_{l-1}, \Omega_{l-1}^{c_i})\} I_t(c_i, c) & \text{if} c = V(x_l) \\ \sum_{i=1}^{M} \Lambda(c) N_0(X_{l-1}, \Omega_{l-1}^{c_i}) I_t(c_i, c) & \text{else} \end{cases} \quad (3)$$

最終的に生成可能な単語系列集合中の，文 X と距離1である系列の数 $N_1(X)$ は次式で求まる．

$$N_1(X) = \sum_{i=1}^{M} N_1(X_L, \Omega_L^{c_i}) I_t(c_i, c_e) \quad (4)$$

$N_1(X)$ を求める具体的な計算手順をpascal風に記述したものを図3に示す．最も計算量の多い図3のStep 3について計算量をみると，いちばん外側のループが $L-1$ 回，1つ内側のループが M 回，最も内側のループが最大 M 回で，このアルゴリズムの1文当りの時間計算量は $O(LM^2)$ である．

4 単語認識率と文認識率の関係

論説文136文を形態素解析し品詞の列に変換したものより，品詞のbigramとtrigramの有無を求めた．単語数は496語で，カテゴリとして，文節構造を表すオートマトン(2)より抽出した67品詞を設定した．そして136文それぞれについて，提案したアルゴリズムによって距離1の文の数を求め，我々の提案した評価式によって単語認識率と文認識率の関係を推定した．併せて136文の文認識のシミュレーションも行った．

推定値とシミュレーションの結果を図4に示す．推定値よりシミュレーションの方がかなり高い文認識率を示している．設定した品詞が多く平均文長9.8単語と長いため，距離1の文の数が多く，評価式の精度が落ちてしまったと考えられる．しかし，bigram,trigram双方の相対的な傾向は十分示されている．

5 まとめ

品詞等のカテゴリのbigramやtrigramの有無を用いた文認識系の単語認識率と文認識率の関係を推定するための，距離1の文の数を計算するアルゴリズムを提案した．1文に対する距離1の文の数を計算するのに要する計算量は，長さを L，カテゴリの種類数を M とすると $O(LM^2)$ である．

参考文献

(1) 大槻他: 音講論,2-Q-6(1992-10)

(2) 伊藤他: 信学技報,SP87-104(1987-12)

```
Step 1 :(M:カテゴリの種類数, L:長さ)
  for all 1 < i < M, 1 < l < L do
    N0(Xl, Ωl^ci) := 0
    N1(Xl, Ωl^ci) := 0
Step 2 :
  for i := 1 to M do begin
    if ci = V(x1) then begin
      N0(X1, Ω1^ci) := It(ce, ci)
      N1(X1, Ω1^ci) := It(ce, ci) × (Λ(ci) − 1)
    end
    else N1(X1, Ω1^ci) := It(ce, ci) × Λ(ci)
  end
Step 3 :
  for l := 2 to L do begin
    for i := 1 to M do begin
      if ci = V(xl) then begin
        N0(Xl, Ωl^ci) := 1
        for j := 1 to M do begin
          N1(Xl, Ωl^ci) := N1(Xl, Ωl^ci)
            +{N1(Xl−1, Ωl−1^cj) + (Λ(ci) − 1)
            ×N0(Xl−1, Ωl−1^cj)} × It(cj, ci)
        end
      end
      else begin
        N0(Xl, Ωl^ci) := 0
        for j := 1 to M do begin
          N1(Xl, Ωl^ci) := N1(Xl, Ωl^ci)
            +It(cj, ci) × N0(Xl−1, Ωl−1^cj) × Λ(ci)
        end
      end
    end
  end
Step 4 :
  N1(X) := Σ_{i=1}^{M} N1(XL, ΩL^ci) × It(ci, ce)
```

図3: 距離1の文の数を計算するアルゴリズム

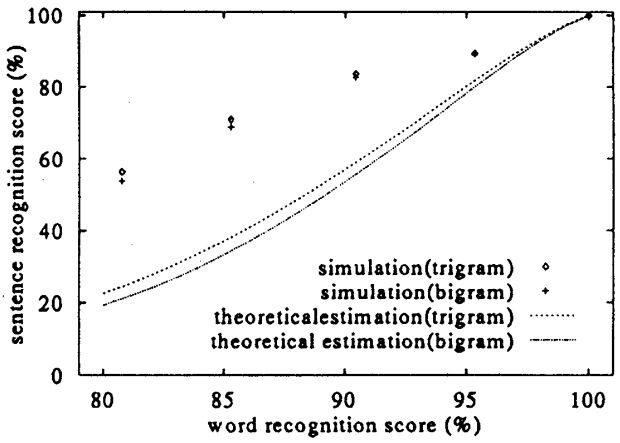

図4: bigram,trigramを用いた文認識における単語認識率と文認識率の関係(論説文136文)