知識・自然言語処理 or 自然言語処理　参考資料

品詞bigram(Bi-POS)モデルを用いたコスト最小法による形態素解析は、以下に示されるビタービアルゴリズムによって効率的に計算される。

記号の定義：

長さLの入力文字列：$S_L = c_1, c_2, ..., c_L$

出力単語列：$W = w_1, w_2, ..., w_{|W|}$　（すべてのw_iは辞書に入っている単語であること）

出力品詞列：$T = t_1, t_2, ..., t_{|T|}$　（ちなみに$|W| = |T|$である。）

単語w_iの品詞：$t(w_i)$　　単語列Wの品詞列：$t(W) = t(w_1), t(w_2), ..., t(w_{|W|})$

品詞t_{i-1}の直後に品詞t_iが続くコスト(品詞バイグラムのコスト)：$con(t_{i-1}, t_i)$

品詞t_iの場合に単語w_iが出現する単語コスト：$word(w_i, t_i)$

解くべき問題(品詞bigramモデルを用いたコスト最小法による形態素解析の形式的定義)：

$$(\hat{W}, \hat{T}) = \arg\min_{W,T} \sum_{i=1}^{|W|} \{con(t_{i-1}, t_i) + word(w_i, t_i)\}$$　（ここでt_0は「文頭」という特殊な品詞）

上記m問題のビタービアルゴリズムによる計算方法：

(0) 記憶領域（変数）

$\#w(x)$：文字位置xで終わる単語の数（簡単のためすべての位置で1以上とする）

$w_x(j)$：文字位置xで終わる単語のうちj番目の単語

$t_x(j)$：$w_x(j)$の品詞

$\delta_x(j)$：$w_x(j)$を採用した場合の先頭から$w_x(j)$までの最小コスト

$B_x(j)$：バックポインタ($\delta_x(j)$を達成するために選ばれた左隣の単語)

(1) 初期化：

・文頭部分の初期化：$\#w(0) = 1$，$t_0(1) =$文頭，$\delta_0(1) = 0$

・辞書を用いて単語ラティスを作る　($\#w(x), w_x(i), t_x(i)$の設定)

(2) 繰り返し：　$x = 1: L$

for $i = 1: \#w(x)$

$$y = x - |w_x(i)|$$

$$\delta_x(i) = \min_{j=1: \#w(y)} \{\delta_y(j) + con(t_y(j), t_x(i)) + word(w_x(i), t_x(i))\}$$

$$B_x(i) = \arg\min_{j=1: \#w(y)} \{\delta_y(j) + con(t_y(j), t_x(i)) + word(w_x(i), t_x(i))\}.$$

(3) 終了：

$$bp(0) = \arg\min_{i=1: \#w(L)} \delta_L(i)$$

$$w'_1 = w_L(bp(0)),\ left(1) = L - |w'_1|,\ bp(1) = B_L(bp(0))$$

もっと簡潔に書けるが、Tri-POSモデルのときにこの方が考えやすい。

(4) バックトラック

$$\left.\begin{aligned} w'_k &= w_{left(k-1)}(bp(k-1)) \\ left(k) &= left(k-1) - |w'_k| \\ bp(k) &= B_{left(k-1)}(bp(k-1)) \end{aligned}\right\} \quad k = 2, 3, ..., M \quad (left(M) = 0)$$

最後にw'_kを逆順にして出力：$\hat{W} = \hat{w}_1, \hat{w}_2, ..., \hat{w}_M$　($\hat{w}_i = w'_{M-i+1}$)

(品詞列は $w \rightarrow t$ と置き換えればよい)

以上