ブラウジング支援のための一覧性の高いキーワードリストの抽出

上村卓史† 喜田拓也† 有村博紀†
†北海道大学大学院情報科学研究科

1 はじめに

ウェブ上のコンテンツに対するタグ付けが近年注目されている*†．これらのサービスは情報への効率的なアクセスをユーザに提供する．しかし一方で，このようなサービスに対応していないページが現在でも大多数である．本稿では，ウェブページに対して半自働的にタグ付けを行うことによるブラウジング支援 [1] を実現するための新たなキーワード抽出アルゴリズムを提案する．

本稿で提案するアルゴリズムは，与えられた文字列に対し，ある単語数以下の全ての単語 N グラムを表す索引構造である単語 N グラム木 [2] を用いることで，テキスト中の部分文字列の同値関係 [3] を考慮し，冗長なものを候補から除外してキーワードを列挙することができる．また，ユーザにとってより一覧性の高いキーワードのリストを出力するため，得られたキーワード集合からさらに選別を行う手法を提案する．

2 準備

アルファベットを Σ とする．Σ 上の文字列全体の集合を Σ^* で表す．文字列 $x \in \Sigma^*$ の長さを $|x|$ と表す．特に長さが0の文字列を**空語** (*empty word*) といい，ε で表す．Σ^+ を $\Sigma^* \setminus \{\varepsilon\}$ と定義する．文字列 $x, y \in \Sigma^*$ の連結を $x \cdot y$ で表す．**単語** w を $w \in \{W \cdot \# | W \in \Sigma^+\}$ とする．$\#$ は区切り記号であり，$\# \notin \Sigma$ である．以降では，文字列 S は単語の並び $S = w_1 \cdots w_m$ であると仮定する．このような文字列を特に**単語列**と呼ぶ．ここで，w_m は $w_1 \cdots w_{m-1}$ に現れない単語とする．**単語列** X が $S = w_1 \cdots w_m$ **中に位置** i **で出現する**とは，$X = w_i \cdots w_j (1 \le i, j \le m)$ である i, j が存在することをいう．また，S 中における X の**出現回数**とはそのような位置 i の個数であり，$f(T, X)$ と書く．ただし，$X = \varepsilon$ ならば $f(T, X) = \infty$ と定義する．

ある単語列 $T = w_1 \cdots w_m$ に対して，$w_i \cdots w_j (1 \le i, j \le m)$ を T の**部分単語列**と呼ぶ．ある整数 $N > 0$ に対し，文字列 $T = w_1 \cdots w_m$ の**単語 N グラム集合** $WS(T, N)$ とは，単語数が N 以下の T の全ての部分単語列の集合，即ち $\{w_i \cdots w_j \mid 1 \le i \le m, j \le min(m, i+N-1)\}$ である．

ある文字列 T について，X を T の部分単語列とおき，Y を X の部分単語列 $Y = X[i \ldots |Y|] (0 \le i \le |X| - |Y|)$ とおく．このとき，S 中で Y が出現するすべての位置 j について $S[j-i \ldots j-i+|X|] = X$ であるとき，Y は X と**同値**であるといい，$X \equiv Y$ と書く．次に，$[X]_{\equiv}$ を X と同値であるすべての S の部分単語列の集合とする．$\overleftrightarrow{X}$ を $[X]_{\equiv}$ の中で最長の単語列とする．単語列 $T = w_1 \cdots w_m$ の **N 単語以下のキーワード候補集合** $K(T, N)$ を，$\{X | X \in WS(T, N) \text{ and } X = \overleftrightarrow{X}\}$ と定義する．

単語列 X	情	報	抽	出	に	関	す	る	論	文
品詞	名詞		名詞		助詞	動詞			名詞	
品詞スコア	10		10		0	2			10	
単語の長さ	4		4		1	4			4	
単語スコア	40		40		0	8			40	

品詞スコア	
品詞	スコア
名詞	10
動詞	2
形容詞	2
その他	0

図 1: 品詞スコアの例と，単語列 X = "情報抽出に関する論文" に対する単語スコアの計算例．

3 キーワードに対するスコア付け

ある $N > 0$ に対し，単語列 $T = w_1 \cdots w_m$ のキーワード候補 X とは $K(T, N)$ の要素である．まず，X 中の各単語 w の品詞から，あらかじめ定められた品詞スコア $C(w)$ をそれぞれ求める．次に，品詞スコアと単語の長さの積 $C(w) \cdot |w|$ を，**単語スコア** $W(w)$ とする．このとき，キーワードに含まれる各単語の単語スコアの総和 $\Sigma_{i=1}^{m} W(w_i)$ を**単語列スコア** $P(X)$ とする．最後に，X の T 中の出現回数の log をとった値と単語列スコアの積から，$P(X) \cdot \log f(T, X)$ を，キーワードに対する最終的なスコアとする．キーワードに対する品詞スコアの例および単語スコアの計算例を図 1 に示す．

ここで，本稿におけるキーワード抽出を，次のように定義する．

単語 N グラムキーワード抽出問題: 与えられた入力テキスト T に対し，N 単語以下のキーワード候補集合のうち，上位 k 個をスコアの降順で出力せよ．

4 単語 N グラム木によるキーワード抽出

与えられた長さ n，単語数 m の単語列 $T = w_1 \cdots w_m$ と整数 $N > 0$ に対し，T の**単語 N グラム木** $WST(T, N)$

Keyword Extraction for Browsing Support
†Takashi Uemura,†Takuya KIDA and†Hiroki ARIMURA
†Graduate School of Information Science and Technology, Hokkaido University
*はてな，URL: http://www.hatena.ne.jp/
†Flickr，URL: http://www.flickr.com/

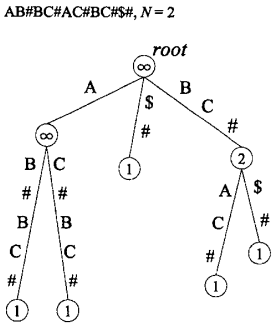

図 2: 単語列 $T = AB\#BC\#AC\#BC\#\$\#$ に対する単語 2 グラム木.

とは，T の単語 N グラム集合 $WS(T,N)$ を表す圧縮トライである．$WST(T,N)$ の各ノード v には，木の根から v への道にラベル付けされた文字列を連結した文字列の，最も後方にある # までの単語列が対応する．例として文字列 T =AB#BC#AC#BC#$# に対する単語 2 グラム木を図 2 に示す．ノード v の中に記された数字は，v に対応する単語列 X の出現回数 $f(T,X)$ を示す．なお，$N = \infty$ とすると，単語の先頭から始まる T の全ての接尾辞を表す単語接尾辞木 [4] と同形の木となる．

単語 N グラム木を用いると，各ノードを深さ優先探索することで，ノードに対応する単語列 X をキーワード候補として列挙することができる．またこのとき，成澤らの手法 [3] を用いて $X = \overleftrightarrow{X}$ であるかを判定することで，$K(T,N)$ の要素のみから上位のキーワードをすべて出力することができる．

5 キーワード抽出結果の再選別

前節の抽出アルゴリズムでは，抽出されたキーワード間に同じ単語が複数現れる場合があるなど，冗長性が残っている．そこで，スコアが上位のキーワードから順に以下の処理を行い，冗長なキーワードを排除することで一覧性を向上させる．

L を単語の集合とする．初期状態では，L は空とする．キーワード $X = w_i \cdots w_j$ について，単語の多重集合 $\overline{L}_X = \{w_l | i \leq l \leq j, w_l \notin L\}$ を求める．ここで，$\overline{L}_X$ の要素数を $|\overline{L}_X|$ と置く．このとき，ある閾値 $\delta_1, \delta_2 (0 < \delta_1, \delta_2 < 1)$ に対し，

$$\frac{|\overline{L}_X|}{|X|_{word}} \geq \delta_1 \text{ かつ } \frac{\sum_{w \in \overline{L}_X} W(w)}{\sum_{n=i}^{j} W(w_n)} \geq \delta_2$$

であるならば，X をキーワードとして出力し，X に含まれる全ての単語を L に追加する．以上の処理を上位のキーワードから順に行う．

表 1 に，青空文庫*より入手した芥川龍之介の「鼻」の本文に対するキーワード抽出結果と，$\delta_1 = \delta_2 = 0.5$ として選別したときの上位 10 個のキーワードを示す．ただし，形態素解析に MeCab0.96†を，品詞スコアに図 1 の値を用いた．また，1 文字が表す情報の量の違いを考慮し，漢字はかな文字の 2 倍の長さとして扱った．δ_1, δ_2 は予備実験により求めた値を用いた．選別前は長いキーワードの部分単語列が多く現れているが，選別後は,「弟子の僧」と「池の尾」や,「内供」と「禅智内供」のように，単語の重複を許しつつ明らかな重複を取り除くことができており，自然な結果が得られているといえる．

表 1: 芥川龍之介「鼻」に対するキーワード抽出結果上位 10 個と選別後の結果上位 10 個.

選別前のキーワード	選別後のキーワード
弟子の僧	弟子の僧
弟子の僧の	内供
内供	鼻
鼻	禅智内供
供	自分
弟子	池の尾
弟子の	顔
弟子の僧は	中童子
内供は	上唇の上から顔の下まで

6 おわりに

本稿ではブラウジング支援のための単語 N グラムキーワード抽出アルゴリズムを提案した．今後の課題として，抽出品質の定量的な評価と，ブラウジング支援への適用における評価が挙げられる．

参考文献

[1] 松尾豊, 福田隼人, 石塚満. ユーザ個人の閲覧履歴からのキーワード抽出によるブラウジング支援. 人工知能学会論文誌, Vol. 18, No. 4E, pp. 203–211, 2003.

[2] 上村卓史, 喜田拓也, 有村博紀. ウェブ閲覧における効率的なキーワード抽出とその利用. データベースと Web 情報システムに関するシンポジウム (DBWeb2007), 2007.

[3] 成澤和志, 稲永俊介, 坂内英夫, 竹田正幸. 接尾辞配列による効率的な文字列上の同値類計算. 電子情報通信学会技術研究報告. COMP, コンピュテーション, Vol. 107, No. 24, pp. 63–70, 2007.

[4] S. Inenaga and M. Takeda. On-line linear-time construction of word suffix trees. In *Proc. of 17th Ann. Symp. on Combinatorial Pattern Matching*, pp. 60–71, 2006.

*URL: http://www.aozora.gr.jp/

†URL: http://mecab.sourceforge.net/