電子文書参照支援システムBENLIにおける内容検索機能
— 前処理としてのキーワード検索 —

3F−8

石橋 由希夫，和田 浩一，小川 洋一
セコム(株) IS研究所

1 はじめに

我々は，電子文書参照支援システムBENLIの研究／作成を進めている[1]．BENLIにおける内容検索とは，ユーザが入力した自然言語文に対して，その文と検索対象の文の意味的な比較を行なう[2]ことで類似する部分を検索する機能である．しかし，対象文書の全てについて比較を行なうとシステム全体の負荷が大きくなり過ぎると考えられる．そこで，前処理としてキーワード検索を行なうことにより比較すべき部分を絞り込むという方法をとることとした．そのために，あらかじめ設定した検索件数が得られるような検索式を生成しなければならない．本稿では，一般のサーチャーがとっている戦略[3, 4, 5]を取り入れた，検索式生成のためのアルゴリズムを提案する．

2 キーワード検索の概要

検索対象となる文書はハイパーテキストとして格納される．文書は構造的なまとまりごとにノード（以後，“文書ノード”と呼ぶことにする）として表される．この文書ノードを検索の単位とする．また，各文書ノードにはID番号が与えられている．

キーワード検索では，ユーザが検索キーとして入力した文を解析して得られた意味構造中の自立語をキーワードとみなして，それらを論理結合した検索式で検索する．その検索式は，検索件数が前もって設定しておいた範囲になるように生成される（以後，このような検索式を“適切な検索式”と呼ぶことにする）．そして，適切な検索式による検索の結果得られた文書ノードのID番号のリストをキーワード検索の結果とする．検索には，キーワードからそのキーワードが出現する文書ノードのIDのリストを得ることができるキーワード索引を用いる．

3 概念とその特定性の高さ

概念とその特定性の高さを以下のように定義する．

(1) 概念

ある自立語に対して，その自立語に意味的に類似している自立語の中で，上位概念語または下位概念語にあたるものを類義語，それ以外の語を同義語と呼ぶことにする．

このとき，自立語とその自立語の同義語や類義語を和結合したものを概念と呼ぶことにする．

あるキーワードkについて，kとその同義語の集合を和結合したものをK，また，類義語の集合を和結合したものをK^+と表す．

(2) 概念の特定性の高さ

キーワードの論理結合式Eによる検索件数を返す関数を，$docfreq(E)$としたとき，概念Kの特定性の高さを，

$$S = \frac{1}{docfreq(K)}$$

(ただし，$docfreq(K) = 0$のとき$S = 0$)

で表すことにする．

4 適切な検索式を生成するためのアルゴリズム

入力のN個の自立語から得られた概念を，

$$K_1, K_2, ..., K_N$$

とする．このN個の概念の中に特定性の高いものがある場合，その特定性の高い概念から検索式を生成しても検索候補の絞り込みが十分にできる[3, 5]．そこで，K_iの特定性の高さをS_iとしたとき，与えられた閾値θに対して，

$$S_i > \theta$$

となるiがあれば特定性の高い概念から検索式の生成を行なう（4.1節）．そうでなければ，全ての概念の積から検索式の生成を行なう（4.2節）．

なお，以下の説明で，前もって設定しておいた検索件数の範囲の上限と下限を各々MAX, MINとする．

4.1 特定性の高い概念からの検索式の生成

生成した検索式の検索件数がMAXより多い場合は，その検索式に新たな概念を積結合して，検索件数を減らす．逆にMINより少ない場合は，検索式中の同義語からなる概念に対する類義語の和結合などにより，検索件数を増やす．具体的なアルゴリズムは以下に示す通り．

要素数$rlen$の配列$record$について，

$$\bigvee_{i=1}^{rlen} (record[i])$$

Keyword search as pre-presessing for content search in the BENLI system
Yukio ISHIBASHI,Koichi WADA,Youichi OGAWA
SECOM IS Laboratory

が適切な検索式となるような $record$ を求める．ただし，$i < j$ ならば，$S_i \geq S_j$.

$rlen \leftarrow 1$; {$record$ の要素数}
$i \leftarrow 1$; $record[1] \leftarrow (K_1)$;
loop :
  if $i > N$ then exit ;
  if $MAX < doqfreq(\bigvee_{j=1}^{rlen}(record[j]))$ then
    begin "$record[rlen]$と(K_{i+1})を積結合したものを
      新たに$record[rlen]$に代入する;
      $i \leftarrow i+1$; goto loop end
  else if $MIN > doqfreq(\bigvee_{j=1}^{rlen}(record[j]))$ then
    begin $rlen \leftarrow rlen+1$;
      if "$record[rlen-1]$に(K_i)がある" then
        "$record[rlen-1]$の(K_i)の部分を(K_i^+)に置
        き換えた式を$record[rlen]$に代入する"
      else begin
        "$record[rlen-1]$の(K_i^+)の部分を(K_{i+1})に
        置き換えた式を$record[rlen]$に代入する";
        $i \leftarrow i+1$ end;
      goto loop end

4.2 全ての概念の積からの検索式の生成

生成した検索式の検索件数が MAX より多い場合は，その検索式中の同義語と類義語からなる概念から類義語を除くことで，検索件数を減らす．逆に MIN より少ない場合は，積結合している概念の除去などにより，検索件数を増やす．適切な検索式を $record$ に求める具体的なアルゴリズムを以下に示す．ただし，配列 exp, $under$, $over$ には検索式が代入される．また
$docfreq(\mathtt{nil}) = 0$ とする．

$exp[1] \leftarrow \bigwedge_{i=1}^{N}(K_i \vee K_i^+)$;
$record \leftarrow$ nil; $elen \leftarrow 1$; {exp の要素数}
$olen \leftarrow 0$; $ulen \leftarrow 0$; {$over, under$ の要素数}
loop:
{適切な検索式が生成できない場合}
if $elen = 0$ then
  if $ulen \neq 0$ then begin
    "配列$under$の中で検索件数が最も多いものを
    $record$に格納する"; exit end
  else begin
    "配列$over$の中で検索件数が最も少ないものを
    $record$に格納する"; exit end;
{以下で exp の中に適切な検索式を探す}
$olen \leftarrow 0$; $ulen \leftarrow 0$;
for $i = 1$ to $elen$ do
  if $MAX < docfreq(exp[i])$ then
    begin $olen \leftarrow olen+1$;
      $over[olen] \leftarrow exp[i]$ end
  else if $MIN > docfreq(exp[i])$ then
    begin $ulen \leftarrow ulen+1$;
      $under[ulen] \leftarrow exp[i]$ end
  else if $docfreq(exp[i]) > docfreq(record)$ then
    $record \leftarrow exp[i]$;
if $record =$ nil then
  "$under$の組を調べる"-(1) ;
if $record =$ nil then
  begin "別の検索式を新たにexpに代入する"-(2);
    goto loop end

(1),(2) の手順は以下の通りである．

(1) $under$ の組を調べる．

if $ulen > 1$ then
  for $i = 2$ to ulen do
    "$under$ の要素からi 個取り出して和結合したものの
    中で適切な検索式になるものがあるならば，その中で
    検索件数が最も多い式を$record$ に代入する"

(2) exp に適切な検索式がなかった場合に別の検索式を exp に代入する．

$elen \leftarrow 0$;
for $i = 1$ to $olen$ do
  for $j = 1$ to N do
    if "$over[i]$に$(K_j \vee K_j^+)$がある" then
      begin $elen \leftarrow elen+1$;
        "$over[i]$ の$(K_j \vee K_j^+)$の部分をK_j に
        置き換えた式を$exp[elen]$ に代入する"
      end;
for $i = 1$ to $ulen$ do
  for $j = 1$ to N do
    if "$under[i]$に(K_j)または$(K_j \vee K_j^+)$がある" then
      begin $elen \leftarrow elen+1$;
        "$under[i]$ から(K_j), $(K_j \vee K_j^+)$の部分を
        除いた式を$exp[elen]$ に代入する"
      end

5 おわりに

検索件数が適当な値になるような検索式を生成するためのアルゴリズムを提案した．現在，このアルゴリズムを実現するプログラムを作成中である．また，並行して対象文書の技術用語などを登録したシソーラスの作成も進めている．今後，検索実験によりこのアルゴリズムの評価を行なう予定である．

参考文献

[1] 小川他．電子文書参照支援システム BENLI− システムの構成と機能の概要 −．第 45 回情処全大 *3F-07*, 1992-10.

[2] 和田他．電子文書参照支援システム BENLI における内容検索機能 − 意味構造の一致度の判定手法 −．第 45 回情処全大 *3F-09*, 1992-10.

[3] Oldroyd B.K. Citroen C.L. Study of strategies used in on-line searching. *On-Line Review*, Vol. 1, No. 4, pp. 295–309, 1977.

[4] Marcia J.Bates. Information Search Tactics. *Journal of the American Society for Information Science*, pp. 205–214, 1979.

[5] 三輪眞木子．サーチャーの時代．丸善，1986.