# 複合語マッチングと共起情報を併用する情報検索

山田剛一[†] 森 辰則[†] 中川裕志[†]

情報検索においては，検索対象の規模が拡大するにつれ，検索精度の向上がより強く求められてきている．そこで本論文では，複合語をまとまりとして扱う手法と，単語の共起情報を用いる手法を統合することにより，検索システムの精度向上を図ることを提案する．複合語は全体で1つの概念を表現しており，まとまりとして扱うことが望ましいが，複合語どうしをマッチさせる場合には部分的なマッチングを考慮する必要が生じる．このマッチングを行い文書をスコア付けする手法を考案した．さらに，単語が複合語を構成せずに共起する場合もスコアに反映させるため，共起情報を利用する手法と組み合わせ，評価実験を行ったところ，単語の重みに基づく手法，およびそれに共起情報を加える手法のいずれよりも良い検索精度が得られることが確認できた．

# Information Retrieval Based on Combination of Japanese Compound Words Matching and Co-occurrence Based Retrieval

KOICHI YAMADA,[†] TATSUNORI MORI[†] and HIROSHI NAKAGAWA[†]

To improve retrieval efficiency in information retrieval of Japanese documents, it is necessary to use the information about the structure of compound words because, in Japanese, they are frequently used and describe a concept as a whole. It is also nccessary to use the information of co-occurrence of words because the compound word in a query might appear independently in a document. In this paper, we propose a new ranking method, which is combination of two methods. Namely, the method using information of co-occurrence, and the method using information of making compound of words. We evaluate the proposed method by the public test collections. Our experimental result shows that the method we proposed improves both recall and precision as compared with the traditional vector space model based on $tf \cdot idf$ method, and also improves them as compared with the method which use information only about co-occurrence of words.

## 1. はじめに

ネットワークの発展により，一般ユーザが大規模全文データベースに対して検索を行う機会が増えている．しかし，現在も広く使用されている完全一致モデルでは，データベースの規模が大きくなると文献の絞り込みが素人の手には負えなくなる．これに対し，古くから部分一致モデルが提案されている[1),2)]．これは単語の統計情報を利用して，文書をユーザの要求に対する類似度でランクづけする手法である．これによりユーザへの技術的負担は軽減されるが，肝心の検索エンジンの精度が高くなければ，大規模データベースにおける検索で正しい文書を上位にランクさせることができず，やはりユーザに負担を強いることになる．しかし，従来の単語の統計情報を利用した手法[2)]では，言語の持つ統語的な情報を考慮に入れずに類似度計算を行うため，必ずしも満足のいく結果が得られない．そこで，本論文では検索用のキーワードとして複合語を用いることにより，概念のまとまりをとらえ，より適切な類似度を求める手法を提案する．

複合語は，日本語において非常に多く現れ，またつねに新しく生成されている．表層上は，複合語は既知の単語の列であるが，全体としては，単一だが複雑な概念を表している．したがって，複合語をまとまりとして扱うことが情報検索システムの精度向上に効果があると期待できる．日本語の複合語の例として，次の組を考えてみる．

「評価システム」「システム評価」

これら2つは同じ名詞から構成された複合語であり，異なるのは語順のみである．にもかかわらず，この2つの複合語は違う概念を表している．前者は何かを評

† 横浜国立大学工学部
Faculty of Engineering, Yokohama National University

価するシステムのことであり，一方，後者は何らかのシステムを評価することである．情報検索システムとしては，この2つを区別すべきである．しかし，従来の情報検索システムでは，いま述べたような複合語の情報を扱うことができない．たとえば，単語の重みに基づいたベクトル空間モデル（VSM）[1]や，それを拡張し語の共起情報を加えたモデル[3)~5)]では，語の出現の順序を考慮していないので前述の例のような語順の違いによる情報を扱うことができない．

複合語に関しては，その文法的内部構造に依存した解析を行う研究もある[6),7)]．文献6)のモデルでは，単語が複合語を構成する際に主辞になりうるかといった情報を用意しておく．このことにより一歩踏み込んだ解析を実現しているが，特別なリソースが必要であるため一般性に疑問が残る．一般に複合語の文法的内部構造を同定することは，構文解析にも似た計算コストの高い処理を要求される．そこで我々は，複合語をその内部構造ではなく，単語がある順序で連接しているという観点からとらえる統計的処理に基づく検索エンジンを提案する．

さて，複合語という構造は統語的に意味のまとまりを表しているものであるが，見方を変えれば，単語の共起の特殊な形態ともとらえることができる．単語の共起という現象全体を見ると，共起する単語の間に最も強い統語的関係があるのが複合語であり，複合語を構成する以外にも，係り受けなどの様々な統語的関係が存在しうる．複合語を構成するよりも大きな関係がとらえられれば，さらに大きな意味のまとまりでの検索が可能になるのであるが，それには意味解析が必要であり，その処理のコストが高いため情報検索システムで扱うのは現実的とはいえない．

一方，情報検索の分野においては，単語の共起を扱うことは早くから行われており，完全一致のモデルにおいては近接演算という枠組みで実用化されている[8)]．また，部分一致のモデルにおいても共起情報をスコアに反映させる手法が提案されている[3)]．これらのモデルでは共起する単語の間の統語的な関係は無視しているが，共起する単語間の文字数や単語数などで大域的な関係を近似的に扱う仕組みを有している．このような関係のとらえ方による情報は，複合語の持つ統語情報と比べると確実性に欠けるものの，処理が簡単であり，付加的な情報として有効である．

そこで我々は，複合語による検索と共起情報を用いる検索とを併用し，その結果を統合する**図1**のシステム構成を考えることにした．それぞれの手法による検索エンジンはモジュール化されており，両モジュールの出力は統合されてシステム全体の検索結果となる（このシステム全体の構成を統合モデルと呼ぶことにする）．

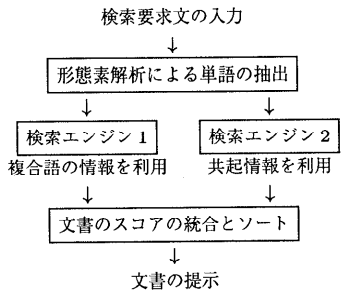



**図1** システムの全体構成
Fig. 1 Diagram of our system.

本論文では，まず複合語の持つ情報を考慮する手法を2章で提案する．我々はこの手法による検索エンジンを用いた検索システムを構築した．これは図1の検索エンジン1に相当する．この実装については3章で述べる．さらに，図1に示す，複合語の情報と共起情報の両者を併用する検索システムを構築し，これをテストコレクションBMIR-J1を用いて評価した．この評価とその結果について，4章で述べる．

## 2. 複合語マッチング

### 2.1 複合語とは

まずここで，本論文で用いる「複合語」の定義をしておく☆．

**定義1**：複合語とは，名詞類に分類される語の列で，他の複合語に含まれないものである．ただし名詞類は，名詞，接頭辞，接尾辞からなる．名詞類である語が助詞「の」を介して連続して存在する場合，これも名詞類の語の列と見なす．

なおこの定義により，名詞類に属する基本単語が孤立して存在する場合には，構成要素数が1である複合語となる．

接頭辞，接尾辞を複合語の構成要素とするのは，それらは名詞と連接し，複合語の概念を構成するのに寄与しているからである．例としては，「非決定性アルゴリズム」の「非」や「性」が接頭辞，接尾辞である．

☆ この定義による「複合語」は，その主辞が名詞とは限らず，また，助詞「の」を含むこともあるため，いわゆる「複合名詞」よりも広い概念である．このため本論文ではこの定義に当てはまるものを便宜上「複合語」と呼ぶが，単に複合してできた語を意味する通常の「複合語」よりもかなり制限されたものであることに注意されたい．

助詞「の」を介しての語の連続は，連接と近い性質を持ち☆，ともに複雑な概念を構成するのに貢献している．そこで本手法では，「の」によって結合された名詞類の列も複合語として扱う．

複合語の構造情報や意味情報を利用すれば，情報検索システムの精度が向上するはずである．しかし，文書内の複合語の種類は非常に多く，さらに，新しい複合語が次々と作られ使用され始めているのが現状である．よって，電子化された辞書にすべての複合語をエントリとして立てるのは不可能である．この理由から，複合語の意味は，辞書に記述された各構成要素の意味から生成する必要がある．しかし，この目的で使用できる，意味素性の記述を持つ辞書はIPAL[9),10)]しか存在せず，IPALは語彙数が少ないため情報検索には対応できない．

また，もし意味素性が記述されている大規模な辞書が存在したとしても，個々の単語の意味から複合語の意味を生成するには複合語内の文法的な構造を解析する必要があり，この解析は構造解析や意味解析をすることになるので計算量が大きい．そのうえ，多くの場合解析結果が曖昧性を持っている[7),11)]．そこで我々は複合語の真の意味を用いる代わりに，表層表現から得られる情報である，複合語を構成する単語とその順序を，複合語の近似的な意味としてとらえることにした．

以下では，検索対象の文書中に現れる複合語と検索要求文に現れる複合語をマッチングする手法について述べる．

### 2.2 複合語どうしのマッチング

検索要求文 $Q$ に含まれる複合語 $C_j^Q$ の集合を $C^Q$ とする．同様に，文書 $D_i$ に含まれる複合語 $C_k^{D_i}$ の集合を $C^{D_i}$ とする．また，$W_s^Q$, $W_t^{D_i}$ は基本単語，すなわち形態素解析システムが単一の語と見なすものとする．定義1より，複合語は名詞類である基本単語の列であり，次のように表現できる．

$$C_j^Q = /W_1^Q/W_2^Q/\cdots/W_n^Q/ \quad (1)$$

$$C_k^{D_i} = /W_1^{D_i}/W_2^{D_i}/\cdots/W_m^{D_i}/ \quad (2)$$

$$C_j^Q \in C^Q,\ C_k^{D_i} \in C^{D_i}$$

（/ は語の境界を表す）

なお，定義1により助詞「の」は連接と同様に扱うため，式(1)，(2)のパターンには現れない．

さて，$C_j^Q$ と $C_k^{D_i}$ をマッチさせるために，これら2つの複合語に含まれる共通のパターンを抽出する．今，$C_j^Q = /A/B/C/D/$, $C_k^{D_i} = /A/B/C/J/$ の場合を考える．ただし，$A, B, C, D, J$ は基本単語を表している．ここで共通のすべてのパターンを数えあげると，$/A/$, $/B/$, $/C/$, $/A/B/$, $/B/C/$, $/A/B/C/$ の6つとなる．これらの中で，1つの基本単語からなるパターン $/A/$, $/B/$, $/C/$ は共通の基本単語の存在を表しているが，連接についての情報はまったくない．2語からなるパターン $/A/B/$, $/B/C/$ では，共通の語が存在しているという情報だけでなく，$A$ と $B$，$B$ と $C$ が複合しているということまでが表現されている．しかしこの場合これでも不十分で，残りのパターン $/A/B/C/$ の方が，$A, B, C$ がこの順番で複合しているという，より多くの情報を表している．つまり，$C_j^Q$ と $C_k^{D_i}$ の複合語マッチングにおいては，$/A/B/C/$ が最大の情報を持っている．よって，マッチする最長のパターン（この場合 $/A/B/C/$）を使うべきであり，また，それで十分である．なぜなら，これに含まれるパターン $/A/B/$, $/B/C/$, $/A/$, $/B/$, $/C/$ がマッチすることは自明であるからである．よって，連続した語がマッチする部分では，マッチする最長のパターンがマッチングの情報としては必要十分である．なお，以後「共通パターン」というのはこのパターンを指すものとする．

まとめると，複合語マッチングにおいては次の制約を満たす必要がある．

**制約1**：抽出される共通パターンは，他のどの共通パターンにも含まれてはならない．

なお，共通パターンは1つとは限らない．たとえば，$C_j^Q = /A/B/C/D/E/$, $C_k^{D_i} = /B/C/E/$ のとき，共通パターン $P(C_j^Q, C_k^{D_i})$ は $\{/B/C/, /E/\}$ である．部分パターンである $/B/$ と $/C/$ は，$/B/C/$ の構成要素であるので無視する．

### 2.3 パターンの重み

検索要求文中の複合語は，文書中の複合語と完全に一致するとは限らず，むしろ部分的にマッチする場合が多い．よって，複合語全体ではなくマッチしたパターンの重みを考える必要がある．本論文では，パターンを重み付けするための手法として $pf \cdot idf$ を提案する．

検索要求文の複合語と，文書中の複合語が部分的にマッチした際のパターンを $P$ とする．パターンは単語の列であるので，次のように表現することができる．

$$P = /W_1/W_2/\cdots/W_w/ \quad (3)$$

まず，パターンの文書内頻度である $pf$（pattern frequency）を導入する．これは単語の文書内頻度 $tf$（term frequency）と同じ考え方によるもので，

☆ これは，「の」自体が特定の意味を表現しているのでも，特定の意味構造を表現しているのでもないためである．なお，「の」の役割についてはいまだに議論があるところである．

$pf^{D_i}(P)$ は文書 $D_i$ におけるパターン $P$ の出現頻度を表す．なお，文書の長さやその内容は文書ごとに異なるので，(パターンに限らず) 文書内頻度は何らかの方法で正規化することが望ましい．この目的で一般的に用いられているのは文書の単語数であるが，本手法では，$pf$を文書中の複合語の語彙数（種類数）で正規化する．この複合語の語彙数は，単語数よりも文書の内容量を反映していると考えられ，実際に単語数を用いるよりも若干良い結果をもたらすことを確認している．

この正規化されたパターンの頻度 $npf$は次のように定義する．

$$npf^{D_i}(P) = \frac{\log_2(pf^{D_i}(P)+1)}{\log_2 length^{D_i}} \quad (4)$$

ただし，$length^{D_i}$ として，文書 $D_i$ における複合語の語彙数を用いる．

もう1つの尺度として，抽出されたパターンの $idf$ (inverse document frequency) を考える．$idf$は次の定義を用いる．

$$idf(P) = \left(\log_2 \frac{\#doc}{df(P)}\right) + 1 \quad (5)$$

ここで，$\#doc$ はコレクション内の文書の数であり，$df(P)$ はパターン $P$ の出現する文書の数である．

さて，これらを用いて，文書 $D_i$ におけるパターン $P$ の重み $pw^{D_i}(P)$ を次のように定義する．

$$pw^{D_i}(P) = \alpha \times npf^{D_i}(P) \times idf(P) \quad (6)$$

この式は $tf \cdot idf$ と同じ形であるが，係数 $\alpha$ を導入した点が異なっている．この $\alpha$ はマッチングの形態に依存するパラメータであり，これは実験的に定めるものとする．

### 2.4 文書のスコア

さて，すでにパターン $P$ の重みを定義したので，最終的に求める，検索要求文に対する文書のスコアを定義する．このスコアは，検索要求文に対しての文書の関連度を与えるものである．まず，検索要求文中の1複合語に対するスコアを定義する．一般に，検索要求文中の1つの複合語に対し，文書中の多くの複合語が部分マッチするので，すべてのマッチしたパターンの重みの総和をとることにする．

いま，$C^{D_i}$ を文書 $D_i$ に出現する複合語の集合とする．文書 $D_i$ 全体において検索要求文中の1複合語 $C_j^Q$ とマッチするパターンの集合 $AllP$ は次のようになる．

$$AllP(C_j^Q, C^{D_i}) = \bigcup_{C_k^{D_i} \in C^{D_i}} P(C_j^Q, C_k^{D_i}) \quad (7)$$

ただし，$P(C_j^Q, C_k^{D_i})$ は，$C_j^Q$ と $C_k^{D_i}$ の共通パターンの集合である．

このパターンの集合 $AllP$ に属する各パターンの重み $pw$ の総和を，検索要求文 $Q$ 内の1複合語 $C_j^Q$ の重み $CScore^{D_i}(C_j^Q)$ とする．

$$CScore^{D_i}(C_j^Q) = \sum_{P_k \in AllP(C_j^Q, C^{D_i})} pw^{D_i}(P_k) \quad (8)$$

検索要求文 $Q$ 内の各複合語 $C_j^Q$ についての重みの総和を，検索要求文 $Q$ 全体に対する文書 $D_i$ のスコア $ComScore^{D_i}(C^Q)$ として次のように定義する．

$$ComScore^{D_i}(C^Q) = \sum_{C_j^Q \in C^Q} CScore^{D_i}(C_j^Q) \quad (9)$$

ただし，$C^Q$ は検索要求文 $Q$ 中の複合語の集合である．

### 2.5 共起情報を用いるモジュールのスコアリング

検索要求文は，これまで述べてきた複合語マッチングに基づくモデルのモジュールのほかに，共起情報を用いるモジュールにも入力される．このモジュールはすでに文献3) で提案されたものであるため，ここでの説明は詳細には立ち入らず，概略だけにとどめる．

この手法は，単語の重みに $tf \cdot idf$ の値を用いるベクトル空間モデルをベースに，単語の共起情報を盛り込んだものである．具体的には，ある単語から見た別の単語の共起重要度を定義し，これを用いて単語の $tf$を補正することにより，従来の枠組みを維持しつつ共起情報を扱っている．ただし，この手法では検索要求文に出現する単語のみでベクトル空間を構成しており，その点で通常のベクトル空間モデルとは異なっている．

共起重要度は，共起する距離（文字数）が近いほど大きな値をとる近接出現係数，データベース内での出現総数を基準とし共起する回数がそれに近いほど大きな値となる共起係数，および共起する相手の語の $idf$ の，以上3つの積で定義される．共起する距離には閾値 $d_{th}$ が設定されており，$d_{th}$ 文字以上離れて出現する場合には共起と見なさない．評価で用いた $d_{th}$ の値については，4章で述べる．

### 2.6 各モジュールによるスコアの統合

統合システムの出力を得るには，各モジュールが文書に与えるスコアを集計し，統合システム全体における各文書のスコアを計算する必要がある．ここでは2つのシステムの出力するスコアの線形和をとることとし，その比 $\beta$ をパラメータとして最適な比率を求め

ることとする．

$$DocScore^{D_i}(C^Q) = ComScore^{D_i}(C^Q) + \beta \cdot CoScore^{D_i}(C^Q) \quad (10)$$

ただし $CoScore^{D_i}(C^Q)$ は，共起情報を用いるモジュールが出力する，検索要求文 $Q$ 中の複合語の集合 $C^Q$ に対する文書 $D_i$ のスコアである．データベース内の文書は，この検索要求文 $Q$ に対する統合されたスコア $DocScore^{D_i}(C^Q)$ に基づきランクづけする．

## 3. システムの実装

この章では，2章で述べた複合語マッチングのアルゴリズムに基づく検索エンジンの実装方法について述べる．この検索エンジンを効率良く実装するためには，逆引き辞書☆の設計が重要である．

なお，共起情報を扱うモジュールの実装は複合語を扱うほどの複雑さはないため，ここでは取り上げないこととする．文献3)を参照されたい．

我々は2種類の逆引き辞書を使用する（それぞれ，逆引き辞書1，逆引き辞書2と呼ぶことにする）．この構造の例を**図2**と**図3**に示す．

逆引き辞書1のエントリは単語である．単語を与えると，その単語を含む複合語とその出現文書IDが得られる．たとえば，図2では，単語 $A$ を与えると複合語 $/A/W/$ と $/X/A/$ が得られる．そして，$/A/W/$ は文書#5と#23で出現しているという情報も得られる．

逆引き辞書2のエントリは，文書に現れる複合語のすべての部分パターンである．パターンを与えると，その出現文書IDとその文書での出現頻度（$pf$）の組のリストが得られる．図3の例では，$/B/C/$ の出現する文書は#5で，そこでの $/B/C/$ の $pf$ は1となっている．他に出現文書がないことから $df(/B/C/)$ が1であることも同時に分かるので，これらより，文書#5における $pf \cdot idf$ が計算できる．

さて次に，検索要求文 $Q$ 中の複合語の集合 $C^Q$ が $\{/A/B/C/\}$ である場合を例として，本手法でのマッチングアルゴリズムを示す．

**マッチングアルゴリズム**

**step1** 検索要求文 $Q$ 中の各複合語 $C_j^Q$ についてstep2以降を行う．この繰返しは式(9)の総和をとる過程に相当する．

**step2** 逆引き辞書1（図2）を用いて，複合語 $C_j^Q$ を構成する各単語から，複合語 $C_j^Q$ と部分マッチするデータベース内のすべての複合語の集合 $C^{allD}$ を得る．この $C^{allD}$ の各複合語 $C_k^{allD}$ について，step3以降を行う．

例 検索要求文 $Q$ 中の複合語の集合 $C^Q$ の中の1つの複合語 $C_1^Q = /A/B/C/$ に着目する．逆引き辞書1を用いることにより得られるのは $C^{allD} = \{/A/W/, /X/A/, /B/Y/, /B/C/Z/\}$ であり，各パターンはその出現文書へのポインタのリストを持っている（図2）．

**step3** 複合語 $C_j^Q$ と複合語 $C_k^{allD}$ との共通パターンを抽出する．制約1により，抽出すべき共通パターンは他の共通パターンに含まれていないものに限られる．そこで，検索要求文中の複合語に含まれる各パターンが文書中の複合語に含まれるかを調べる際に，検索要求文中の複合語に含まれる最長のパターンから調べはじめる．

例 $/A/B/C/$ のすべての部分パターンは $\{/A/B/C/, /A/B/, /B/C/, /A/, /B/, /C/\}$ である．まず，最長のパターンである $/A/B/C/$ が含まれているかどうかを調べる．$C_4^{allD} = /B/C/Z/$ の場合，$/A/B/C/$ は含まれていない．同様に，次の $/A/B/$ も含まれていない．その次の $/B/C/$ は $/B/C/Z/$

| *entry field* | *link field* | |
|---|---|---|
| simple word | list of compound word (with doc.#) | |
| $A$ | $/A/W/$ | at doc.#5, #23 |
| | $/X/A/$ | at doc.#23 |
| $B$ | $/B/Y/$ | at doc.#5, #61 |
| | $/B/C/Z/$ | at doc.#5 |
| $C$ | $/B/C/Z/$ | at doc.#5 |
| ⋮ | ⋮ | |

**図2** 逆引き辞書1
Fig. 2 Inverted file 1.

| *entry field* | *link field* | |
|---|---|---|
| pattern | list of (doc.#,$pf$) pair | |
| $/A/$ | (#5, 8), (#23, 5) | $\rightarrow df(/A/) = 2$ |
| $/B/$ | (#5, 2), (#61, 4) | $\rightarrow df(/B/) = 2$ |
| $/B/C/$ | (#5, 1) | $\rightarrow df(/B/C/) = 1$ |
| ⋮ | ⋮ | |

**図3** 逆引き辞書2
Fig. 3 Inverted file 2.

☆ これは通常「転置ファイル」(inverted file) と呼ぶものであるが，本論文で提案する構造は「転置」と呼べるものではないため，より一般的に「逆引き辞書」と呼ぶことにした．

に含まれているので，/B/C/ は共通パターンとして抽出される．残りの部分パターン {/A/,/B/,/C/} は，/B/C/Z/ の残りである /Z/ には含まれていない．これらより，共通パターン $P(/A/B/C/,/B/C/Z/)$ は {/B/C/} となる．

**step4** 抽出したパターンそれぞれについて，複合語 $C_k^{allD}$ が出現する各文書ごとの $pf \cdot idf$ の値を計算する．$pf$ と $df$ の値は逆引き辞書2（図3）より得る．この $pf \cdot idf$ の値は各文書のスコアに加算していく☆．ただし，ある文書に対するスコアの加算は，各パターンにつき1回のみである．

このステップを繰り返す過程は，式(8)における，総和をとる部分に相当する．

例 逆引き辞書1によれば，複合語 $C_4^{allD}$ = /B/C/Z/ は文書#5のみで現れている．パターン /B/C/ の場合，文書#5における $pf$ の値は1であり，$df$ の値は1となる（図3）．これらを用いて，文書#5における $pf \cdot idf$ の値を計算する．

次に，複合語マッチングに基づく検索システム全体の処理の流れについて述べる．これは，1) 文書データベースの構築（indexing），2) 複合語マッチングのメカニズムによる検索，の2つに大きく分けられる．

データベース構築は次の手続きにより行う．

**データベース作成プロセス**

**step1** 各文書を1文ごとに分割する．

**step2** 各文を形態素解析し単語に分割する．各単語には日本語形態素解析システム茶筌（version 1.0 beta 5）[12]による品詞情報が付与される．

**step3** 形態素解析の結果から，品詞情報をもとに基本単語，複合語を抽出する☆☆．

**step4** マッチングアルゴリズムの項で述べた2つの逆引き辞書を作成する．

検索システムは自然言語による検索要求文を入力とする．データベース構築時と同様の処理を行い，検索要求文から基本単語と複合語を抽出する．

文書の検索は次の手続きに沿って行う．

**検索プロセス**

**step1** 検索要求文を形態素解析し単語に分割する．各単語には日本語形態素解析システム茶筌（version 1.0 beta 5）[12]による品詞情報が付与される．

**step2** 形態素解析の結果から，品詞情報をもとに基本単語，複合語を抽出する．

**step3** 各文書に対し複合語マッチングの手法によりスコアを付与し，関連度の順にソートした文書IDのリストを出力する．

☆ 各文書のスコアは検索開始時には0である．

☆☆ 茶筌はコスト計算と最長一致法により，最も適切と思われる候補を出力するようになっている[12]．なお，茶筌における未定義語は名詞として扱っている．また，茶筌による誤解析は修正していない．

## 4. 評価実験

本論文で提案した統合システムを評価するため，次の3つの手法との比較実験を行った．ベクトル空間モデル[2]と，単語の共起情報に基づく手法[3]，および複合語マッチングによる手法の，あわせて3つである．複合語マッチングによる手法については2章で詳しく述べているため，他の2つの手法について簡単に触れておく．

### 4.1 $tf \cdot idf$ に基づく古典的ベクトル空間モデル

ベースラインとして，$tf \cdot idf$ に基づくベクトル空間モデル[2]を用いた．このモデルでは，単語の重みを伝統的な $tf \cdot idf$ の値としている．各文書をベクトル空間モデルに基づき，検索要求文ベクトルと文書ベクトルの cosine の値によってランキングする．

正規化された $tf$ である $ntf$ は次のように定義する．

$$ntf^{D_i}(P) = \frac{\log_2(tf^{D_i}(P)+1)}{\log_2 length^{D_i}} \tag{11}$$

$length^{D_i}$ は文書 $D_i$ における名詞の種類数である．

$idf$ は次の定義を用いた．

$$idf(N) = \left(\log_2 \frac{\#doc}{df(N)}\right) + 1 \tag{12}$$

ただし，$\#doc$ はデータベースの文書数，$df(N)$ は名詞 $N$ が出現する文書の数である．これらによって $tf \cdot idf$ の値を計算し，ベクトル空間モデルにおいて用いる．

### 4.2 $tf \cdot idf$ をもとに共起情報を加えるモデル

2つめは，単語の重みに共起情報を加えたベクトル空間モデル[3]である．これは本論文で提案したシステムにおける共起情報を用いるモジュールそのものであり，この単独モジュールと比較することにより，統合の効果が確かめられることになる．

このシステムでは，前もって定義する距離の閾値 $d_{th}$ よりも短い距離 $d$ で共起した場合のみ共起と見なして重みを調整している．この閾値 $d_{th}$ の最適値は，文献3)の実験によれば50文字であることが示されているので，今回の評価においても，同一条件の50文字を閾値として用いた．

### 4.3 評価結果

各システムを比較評価するため，情報検索評価用

データベースであるBMIR-J1を利用☆した．これは，日本経済新聞600記事と，60の検索要求文，およびその正解集合からなるものである．この正解にはAとBの2つのレベルが設定されているが，本評価ではどちらも正解として扱った．また，BMIR-J1では，記事にタイトル，キーワード，重みがつけられているが，ここでは記事本文だけを使用した．これは，各手法を純粋に比較することを目的としているためである．なお，60検索要求文の中で，本システムが複合語とする語が含まれているものは56文であるが，一般的な評価をするため，すべての検索要求文を利用した．以下で示すグラフにおける適合率/再現率は，60検索要求文すべてについての値を平均したものである．

さて，他の手法と比較する前に，まず複合語マッチングを用いたモジュールにおけるパラメータ $\alpha$ の値を決定しておく．パターンの重みの係数である $\alpha$ の定めかたとしては，パターンの構成語数や，パターンを含む複合語の語数の関数にすることが考えられるが，実験の結果，最も平均適合率の高かったのは以下の定めかたであった．

**1)** パターン $P$ が検索要求文 $Q$ 中の複合語 $C_j^Q$ に完全に一致している場合には，$\alpha$ はある定数とする．

**2)** パターン $P$ が $C_j^Q$ の一部分である場合には，$\alpha = 1$ に固定する．

これを用いて，複合語マッチングのモジュール単体での評価をした結果が**図4**のグラフである．この図では $\alpha$ の値として，平均適合率が最大となる値，およびその周辺の値を用いた場合のグラフを示した．

まずこの結果から，パターン $P$ が $C_j^Q$ に一致する場合の $\alpha$ の値は1.0よりも小さいほうが良いことが分かる．つまり，検索要求文の複合語と，文書中の複合語がちょうどマッチした場合には重みを小さくすべきだという結果となっている．そこで，1.0から0.0までの間の $\alpha$ について適合率/再現率を計算したところ，$\alpha$ の最適値は0.2周辺であることが分かった．

この結果は，$\alpha$ だけに着目すると不自然な結果に見える．たとえば，検索要求文中の1複合語が/情報/検索/システム/であった場合に，マッチするパターンが/情報/検索/システム/全体の場合には $\alpha = 0.2$ という係数がかかり，マッチするパターンが/情報/検索/であった場合には $\alpha = 1$ なのである．しかし一般的な傾向として，長いパターンであるほど *idf* の値が大きく，もともとのパターンの重みが，/情報/検索/よりも/情報/検索/システム/の方が高いと考えられる．つまり，$\alpha = 0.2$ という数字は，*pf*·*idf* によるパターンの重みほどには，長いパターンを優先させるべきではないことを示している．

さて，他の手法との比較評価の結果は**図5**のような結果となった．ただし，統合の際の係数 $\beta$ は最も

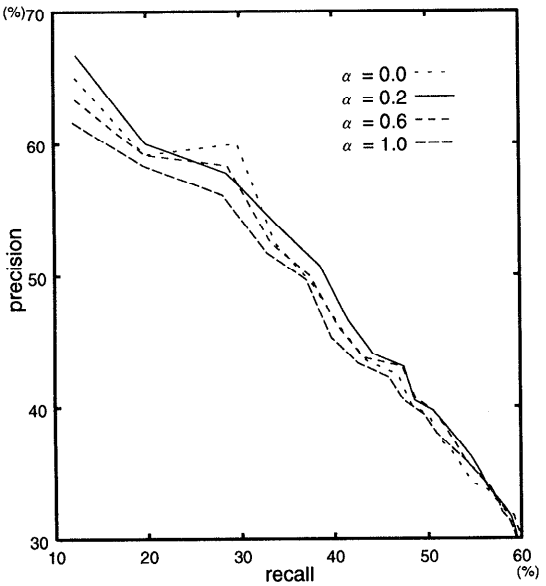



**図4** パラメータ $\alpha$ による適合率/再現率の変化
Fig. 4 Recall/Precision for four values of parameter $\alpha$.

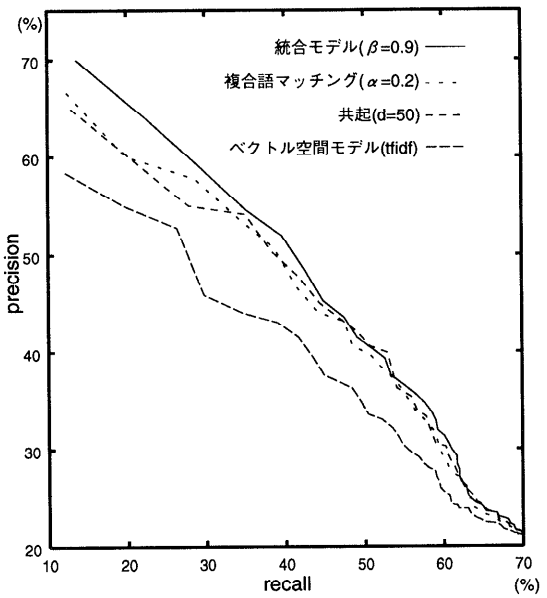



**図5** 他の手法との比較
Fig. 5 Comparison with other algorithms.

☆ 株式会社日本経済新聞の協力によって，社団法人情報処理学会・データベースシステム研究会・情報検索システム評価用データベース構築ワーキンググループが，1993年9月1日から12月31日の日本経済新聞記事を基に構築した情報検索評価用データベース（テスト版）を利用．

平均適合率の高かった $\beta = 0.9$ の場合を示している．このグラフからまず分かることは，統合モデル，複合語マッチングのモデル，共起情報を用いるモデルのいずれも，*tf·idf* の重みに基づくベクトル空間モデルに比べ適合率/再現率とも全般的に向上していることである．このことは，複合語の情報，共起情報のいずれも検索精度の向上に有効であることを一般的に示しているといえる．

複合語マッチングのモデルと共起情報を用いるモデルはほぼ同程度の適合率/再現率を示している．統合モデルはこれらに比較すると全般的に優れているものの，各モジュールによる検索精度の向上分を加算したレベルよりは低くなっている．これは，複合語でマッチする場合に，共起情報のモデルでもある程度のスコアが与えられ，かつ，共起情報のモデルにおいて閾値 $d_{th}$ が50文字と，比較的近い共起のみがスコアに反映するようになっていることによるものと考えられる．逆に，大まかにいえば，複合語マッチングのモデルから単に共起しているという情報を除いた，純粋に意味のまとまりをとらえていることによる効果が，統合モデルと共起のみのモデルとの差として現れていると考えることができる．

なお，この評価では前段階として $\alpha$，$\beta$ の2つのパラメータをテストデータである BMIR-J1 を用いて最適化しているため，クローズ評価となっている．本来は BMIR-J1 の検索要求文を2分割するなどの方法でオープン評価をすべきであるが，BMIR-J1 の検索要求文は絶対量が少ないにもかかわらず種類数が多く，また各検索要求文での複合語の構成語数もまちまちであるため，これを公平に2分割することは困難である．よってオープン評価は，将来，検索要求文の豊富なテストコレクションが公開されるまでの課題としたい．

最後に，参考までに本システムの実行速度を示しておく．ワークステーション（Sun Microsystems 社 SparcStation 20）を用いたところ，BMIR-J1 の1記事をデータベースに登録するのに平均して約60秒，BMIR-J1 の1検索要求文を処理するのに平均して約21秒が必要であった．

## 5. おわりに

本論文では情報検索のための検索モデルとして，複合語マッチングに基づく手法と，共起情報に基づく手法の各モジュールを統合したシステムを提案した．システムの評価においては，テストコレクションの関係でクローズ評価となっているものの，我々の提案した検索モデルが *tf·idf* を単語の重みとするベクトル空間モデルよりも，適合率/再現率とも向上することを示した．また，共起に基づく手法のみによる検索モデルとの比較評価を行ったところ，やはり適合率/再現率とも改善していることが確認された．このことは，語の共起を重みに反映させること，複合語を意味のまとまりとしてとらえることの双方が，適合率/再現率の向上をもたらすことを示している．

今回の手法では，構文解析や意味解析を行わずに意味のまとまりをとらえることにある程度成功したといえるが，これ以上の意味のまとまりを考えるには，やはり複合語の構造解析や，構文解析，意味解析などが必要になってくるので，本論文で提案した手法が軽い処理の限界点に近いのではないかと考えている．

なお，英語において統計的な手法により名詞句の構造解析を行い，その結果を検索に利用する研究[7]があるが，構造を利用しない場合に比べ大きな精度の向上はみられていない．このことは，本手法が処理コストと検索精度のバランスのとれた位置に存在することを示唆しているといえる．

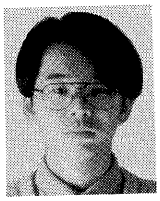
**謝辞** BMIR-J1 を提供してくださった方々，特に（株）リコー小川さん，富士通（株）松井さんに感謝いたします．また，JUMAN，茶筌を公開，発展させ続けている方々に感謝いたします．さらに，査読者の方々には多くの有益なコメントをいただきました．深く感謝いたします．

## 参考文献

1) Salton, G. and McGill, J.M. (Eds.): *Introduction to Modern Information Retrieval*, McGraw-Hill, New York (1983).
2) Frakes, B.W. and B-Yates, R. (Eds.): *Information Retrieval – Data Structures & Algorithms*, P T R Prentice-Hall, NJ (1992).
3) 高木 徹，木谷 強：単語出現共起関係を用いた文書重要度付与の検討，情報処理学会研究報告，96-FI-41-8 (1996).
4) 野口直彦，稲葉光昭，野本昌子，菅野祐司：単語統計情報と言語情報とを併用した新しい文書検索のモデル，情報処理学会研究報告，96-FI-44-5 (1996).
5) 大井耕三，隅田英一郎，飯田 仁：単語間の意味的類似度に基づく文書検索手法，言語処理学会第2回年次大会発表論文集，pp.109–112 (1996).
6) Ogawa, Y., Bessho, A. and Hirose, M.: Simple Word Strings as Compound Keywords: An Indexing and Ranking Method for Japanese Texts, *ACM-SIGIR '93*, pp.227–236 (1993).
7) Evans, D.A. and Zhai, C.: Noun-Phrase Analysis in Unrestricted Text for Information Re-

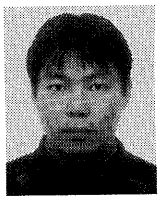
trieval, *Proc. 34th Annual Meeting of the Association for Computational Linguistics* (1996).
8) 大山敬三：インターネットに適応した全文データベース検索システムの構成，学術情報センター紀要第7号，学術情報センター (1995).
9) 情報処理振興事業協会技術センター：計算機用日本語基本動詞辞書IPAL (Basic Verbs) (1987).
10) 情報処理振興事業協会技術センター：計算機用日本語基本名詞辞書IPAL (Basic Nouns) (1996).
11) Hisamitsu, T. and Nitta, Y.: Analysis of Japanese Compound Nouns by Direct Text Scanning, *Proc. 16th International Conference on Computational Linguistics* (*COLING 96*), Vol.1, pp.550–555 (1996).
12) 松本裕治，今 一修，山下達雄，北内 啓，今村友明：日本語形態素解析システム『茶筌』version 1.0b5使用説明書，松本研究室，奈良先端科学技術大学院大学 (1996).

(平成 9 年 10 月 8 日受付)
(平成 10 年 6 月 5 日採録)



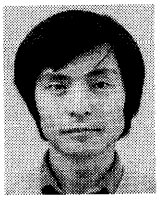
**山田 剛一**（学生会員）

1971年生まれ．1995年横浜国立大学工学部卒業．現在，同大大学院工学研究科博士課程在学中．情報抽出，情報検索，マルチメディア検索などの研究に従事．



**森 辰則**（正会員）

1964年生まれ．1986年横浜国立大学工学部卒業．1991年同大大学院工学研究科博士課程修了．工学博士．1989年から1991年まで，日本学術振興会特別研究員（計算機工学）．1991年より横浜国立大学工学部勤務．現在，同助教授．計算言語学，自然言語処理システム，ディジタルドキュメントなどの研究に従事．



**中川 裕志**（正会員）

1953年生まれ．1975年東京大学工学部卒業．1980年同大大学院博士課程修了．工学博士．1980年より横浜国立大学工学部勤務．現在，同教授．日本語の意味論，語用論，電子化マニュアル検索システム，マルチメディア検索，情報検索，自動ハイパーテキスト化などの研究に従事．