自然言語入力と目次との柔軟な照合による図書検索システム

白 木 伸 征[†,☆] 黒 橋 禎 夫[††]

現代の高度情報化社会では，情報検索はその重要性を高めており，その一分野である図書検索も電子図書館などの普及にともない重要となってきている．これまでの図書検索は，図書のどのような情報を用いるかにより次の 4 種類に分けられる：1. 書名，2. 索引やキーワード，3. 目次，4. 本文テキスト．このうち 3 の目次情報を用いる方法は質的・量的にも理想的であり，目次情報を用いた図書検索システムとして目次の階層構造を利用したものがある．本研究では，それをさらに発展させた図書検索システムを提案する．精度向上のために，目次の階層構造に加えて情報検索で広く用いられてきたベクトル空間モデルと，自然言語入力と目次との柔軟な照合を行う．岩波新書の 1211 冊の図書と 202 個の検索式での評価実験を行い，提案手法の有効性を示す．

Book Retrieval System Based on Flexible Matching between Natural Language Queries and Tables of Contents

Nobuyuki Shiraki[†,☆] and Sadao Kurohashi[††]

Recent information-oriented society needs high-performed information retrieval(IR) systems. Book retrieval system is one of the important application of IR systems. Information exploited by book retrieval systems can be classified as follows: 1. titles, 2. indexing words and keywords, 3. tables of contents, 4. full texts. Among these, 3. tables of contents are the most ideal both in quality and in quantity and there has been a study of book retrieval system utilizing the structure of tables of contents. This paper proposes a new book retrieval method of using the Vector Space Model and flexible matching between natural language queries and tables of contents in addition to using the structure of tables of contents. We report experimental evaluation of the system using 1211 books of Iwanami expository series and 202 queries.

1. はじめに

現代の高度情報化社会において，情報検索は重要な位置を占めるようになってきている．その一分野である図書検索は，電子図書館などの普及により高精度なシステムが必要となっている．

これまでに用いられてきた図書検索の方法は，図書のどのような情報を用いるかにより，次のように分けることができる．

(1) 検索語と書名との一致
(2) 検索語と索引やキーワードとの一致
(3) 検索語と目次との一致
(4) 検索語と本文テキストとの一致

(1) の方法は，OPAC などで現在最も一般的に用いられている方法であるが，書名は図書の内容を十分表せているとはいえないため，満足のいく方法とはいえない．(2) の方法は，図書に索引やキーワードがほとんどついていないため実用的であるとはいえず，また著者以外の第三者が図書に適切なキーワードを与えることは非常に困難である．(4) の方法は，図書の本文がほとんど電子化されていないことと，またこれまでに出版された冊子体の図書を電子化することは容易ではないことから，実用的とはいえない．これに対して (3) の方法は，目次が比較的よく図書の内容を表現していること，データ量的にこれまでの冊子体の図書の目次を電子化することも可能であること，などから最も有効な検索手法になると考えられる[1)]．

我々はこれまでに目次の階層構造を用いる図書検索の手法を提案している[2)]．本研究はこの図書検索システムを基本とし，情報検索で用いられてきたいくつかの手法を組み合わせ，さらに自然言語入力と目次との

† 京都大学大学院工学研究科電子通信工学専攻
Department of Electronics and Communication, Graduate school of Engineering, Kyoto University
☆ 現在，株式会社豊田中央研究所
Presently with Toyota Central R&D Labs., Inc.
†† 京都大学大学院情報学研究科知能情報学専攻
Department of Intelligence Science and Technology, Graduate school of Informatics, Kyoto University

柔軟な照合を行うことによる高精度な図書検索システムを紹介する．

2. 本研究の方法

本研究は高い精度の図書検索を実現するために，図書の目次を利用し以下に示す手法を組み合わせた検索システムの方式を提案する．

(1) 自然文による入力
(2) 検索式の拡張
(3) ベクトル空間モデルと wtf・idf
(4) 検索式と目次タイトルとの柔軟な照合
(5) 目次の階層構造の利用

これらの手法についての説明を以下の節で行う．

2.1 自然文による入力

本研究の図書検索システムは自然文による入力を用いる．自然文による入力の利点は以下の 2 点である．

- 従来の図書検索の方法は単語により質問入力を行うために検索意図を表す単語を選択する必要があるが，自然文による入力は検索意図をそのままの形で入力でき，検索者にとって容易である．
- 従来の単語による質問入力は単語の集合としてしか検索意図を表現することができないが，自然文による入力では，単語間の結び付きや関連性を名詞句などの形で取り出して活用することができる．

以下では，質問として入力された自然文を**検索式**と呼ぶ．たとえば自然文による検索式として，

「日本の公害」
「公害による環境への影響」

などが考えられる．

2.2 検索式の拡張

検索における大きな課題として，検索語の拡張の問題がある.「癌」と「がん」,「計算機」と「コンピュータ」などの同義語,「コンピュータ」と「コンピューター」などの表記の揺れ,「公害」と「汚染」などの類義語・関連語，などによって検索語を拡張することで検索精度が高くなると考えられる．しかし，広い分野の同義語や表記の揺れに対する辞書は存在せず，単純に類義語の拡張を行うとむしろ検索精度が悪くなることが報告されている[3),4)]．

そこで本研究では，機械的に行える範囲で最も効果のあるひらがな形とカタカナ形だけによる拡張を行う．すなわち検索語が「癌」であれば,「がん」や「ガン」も検索語であると考えることにする．以下では，ひらがな形とカタカナ形による検索式の拡張を**かな拡張**と呼ぶ．かな拡張をする場合の重み付けに関しては 2.4.2 項で詳しく述べる．

逆に，ひらがなやカタカナで入力された検索語を漢字に変換することも考えられるが，漢字への変換は曖昧性が高いため非常に難しく，本研究では考慮しないこととする．

2.3 ベクトル空間モデルと wtf・idf

情報検索でよく用いられる手法として，ベクトル空間モデルがある．ベクトル空間モデルは，ドキュメントと質問を多次元ベクトルで表現（それぞれ**ドキュメントベクトル**と**質問ベクトル**と呼ぶ）し，それらのベクトル間の類似度（内積や余弦）を求め，類似度が高い順にドキュメントを順序付けする方法である．

ドキュメントベクトルの要素として，単語の重要度を表す指標である tf・idf が一般的に用いられる[5)]．tf（term frequency）はある単語がドキュメント中に現れる回数であり，多く現れる単語が重要であることを表す．また idf（inverse document frequency）は特定の文書に現れる単語が重要であることを表し，

$$idf = \log \frac{N}{df} \tag{1}$$

という式で表される．ここで N はドキュメントの総数で，df は単語が出現する文書の数である．

図書の目次を検索対象とする場合もこの方法を利用することができる．この場合ドキュメントが図書の目次に相当し，そのドキュメントベクトルを以下では**図書ベクトル**と呼ぶ．

tf は単純には目次中の単語の出現回数であるが，様々な重みを考慮することにより検索の精度を上げることができると考えられる．本研究では以下の 3 種類の重みを考慮する．

(1) 名詞の種類による重み付け
名詞には様々な種類があるが，固有名詞は普通名詞と比較して重要であるといえる．

(2) 目次階層の深さによる重み付け
目次はその階層構造により，上位のタイトル☆の方が下位のタイトルよりも重要な意味を持つと考えることができる．そのため，タイトル中に含まれる単語も同様に上位に現れる方が重要であるといえる．

(3) 1 つのタイトル中の文節数による重み付け
1 つのタイトル中の文節数が少ないほど，そのタイトル中に現れる単語の重要度が高いと考えられる．たとえば,「年金について」というタイトルと「国民の年金に対する意識」というタイトルがある場合,「年金」という単語の重要度は

☆ 図書の書名や目次の章名や節名を以下では**タイトル**と呼ぶ．

前者のタイトル中の方が高いと予想される．

これらの 3 種類の重みを考慮した，ある目次におけるある単語の出現頻度を **wtf**（weighted tf）と呼び，次のように計算する．

$$wtf = \sum_i \left\{ w_{noun} \times \frac{1}{(w_{depth} \cdot depth_i + 1)} \times \frac{1}{(w_{bunsetsu} \cdot bunsetsu_i + 1)} \right\} \quad (2)$$

ここで $\sum_i$ は，着目している単語の目次中の各出現について，重みを計算してその総和をとることを意味する．$depth_i$ は単語の i 番目の出現における目次階層の深さ（書名を 0，章名を 1 のように定義する），$bunsetsu_i$ は i 番目の出現を含むタイトルの文節数を表す．また，w_{noun} は名詞の種類に対する重み（これは単語ごとに決まるので，実際には $\sum$ の外側に出すことができる），w_{depth} は目次階層に関する重み，$w_{bunsetsu}$ はタイトルの文節数に関する重みを表す．これらの重みは，4.3 節で示すように実験により最適な値を求める．

この wtf に式 (1) の idf をかけあわせたものを，**wtf・idf** と呼ぶ．また，目次 D 中の単語 T の wtf・idf の値を以下では $W(D,T)$ と表し，これを図書ベクトルの要素とする．

また質問ベクトルについては，検索式を形態素解析して名詞をすべて抜き出し，その名詞が対応するベクトル要素を 1 とし，それ以外の要素が 0 であるとする．

たとえば検索式が「癌の告知と患者」である場合，その中の名詞は「癌」,「告知」,「患者」である．そのとき，質問ベクトル$\boldsymbol{qv}$は次のように表される．

$$\begin{array}{lcccccc} & & \text{癌} & \text{告知} & \text{患者} & & \\ \boldsymbol{qv} = & (\cdots, & 0, & 1, & 1, & 1, & 0, \cdots) \end{array}$$

また，2.2 節で述べたかな拡張を行う場合，質問ベクトルは，

$$\begin{array}{lcccccccc} & & \text{癌} & \text{がん} & \text{ガン} & & & & \\ \boldsymbol{qv} = & (\ \cdots, & 0, & 1, & w_{kana}, & w_{kana}, & & & \\ & & & \text{告知} & \text{こくち} & \text{コクチ} & \text{患者} & & \\ & & & 1, & w_{kana}, & w_{kana}, & 1, & \cdots) & \end{array}$$

と表すことができる．ここで，w_{kana} はかな拡張に対する重みを表し，4.3 節に示すように実験により最適な値を求める．

このようにして計算した図書ベクトルと質問ベクトルの内積が，その図書の検索意図への一致度の指標となる．

2.4 検索式と目次タイトルの柔軟な照合

ベクトル空間モデルに基づく手法は，検索式中の名詞と目次中に分散している名詞との一致を調べるため，非常に条件のゆるい照合であるといえる．これに対して，検索式中の複合名詞や名詞句などがそのまま目次中の 1 つのタイトルに含まれていれば，その図書は検索意図に強く一致していると考えられる．このことを考える場合の問題点は，複合名詞や名詞句が完全一致する場合は少なく，柔軟な照合が必要になるということである．以下では，正規化名詞列というものを導入することによって，名詞句間の柔軟な照合を実現する方法を示す．

2.4.1 検索式と目次タイトルの解析

検索式と目次タイトルの解析は以下のように行う．解析の例として，

「在日アメリカ人の意見－現代の日本の経済とアメリカの政治の腐敗がもたらす影響」

という自然文に対する処理を考える．

(1) 自然文を形態素解析して単語に区切りその品詞を得る．

「在日 アメリカ人 の 意見 － 現代 の 日本 の 経済 と アメリカ の 政治 の 腐敗 が もたらす 影響」

(2) 句点やハイフンなどで分割する．

「在日 アメリカ人 の 意見」
「現代 の 日本 の 経済 と アメリカ の 政治 の 腐敗 が もたらす 影響」

(3) 名詞と並列語以外をすべて除去する．ただし，その結果が名詞 1 語となる場合は，名詞句の柔軟な照合という目的に合わないため，ここでは扱わないものとする．

ここで**並列語**とは，助詞「と」，助詞「や」などの並列を示す語のことである．

「在日 アメリカ人 意見」
「現代 日本 経済 と アメリカ 政治 腐敗 影響」

(4) 単語列の中に並列語が含まれる場合，その単語列を

「$N_n \cdots N_{-2} N_{-1}$と $N_1 N_2 \cdots N_m$」 (3)

として次のような処理を行う（N_i は名詞）．

並列語はその前後の名詞句に関して並列であることを示すが，その名詞句をどのように区切るかは曖昧である．そこで，以下のように単語の類似度を用いて区切り方を決める．

- N_{-1} と N_i $(1 \leq i \leq m)$ との類似度☆を比較し，最も類似度の高い名詞を N_a とする．ただし，最も高い類似度が 0.5 より小さい場合は，区切りの強い手がかりとは考えられないので，経験的に最も大きな区切りを優先し，$N_a = N_m$ とする．
- N_1 と N_i $(n \leq i \leq -1)$ との類似度を比較し，最も類似度の高い名詞を N_b とする．ただし，最も高い類似度が 0.5 より小さい場合は，上と同様に $N_b = N_n$ する．

このように並列語の前後の名詞との類似度を比較することにより，次のように「$\underline{N_b \cdots N_{-1}}$」と「$\underline{N_1 \cdots N_a}$」の名詞列が並列であると見なすことができる．

$$\underline{\cdots N_{b-1}} \left\{ \begin{array}{c} \underline{N_b \cdots N_{-1}} \\ \text{と} \\ \underline{N_1 \cdots N_a} \end{array} \right\} \underline{N_{a+1} \cdots}$$

その結果，単語列 (3) は

「$\underline{\cdots N_{b-1}}\ \underline{N_b \cdots N_{-1}}\ \underline{N_{a+1} \cdots}$」

「$\underline{\cdots N_{b-1}}\ \underline{N_1 \cdots N_a}\ \underline{N_{a+1} \cdots}$」

の 2 つの名詞列で表現することができる．

例文では「現代 日本 経済 と アメリカ 政治 腐敗 影響」が並列語を含んでいるため，その区切りを決める.「と」の直前の名詞「経済」に最も類似した名詞は「政治」であり,「と」の直後の名詞「アメリカ」に最も類似した名詞は「日本」である．そのため,「日本 経済」と「アメリカ 政治」が並列であると判断されるので，最終的に得られる名詞列は，

「在日 アメリカ人 意見」

「現代 日本 経済 腐敗 影響」

「現代 アメリカ 政治 腐敗 影響」

となる．

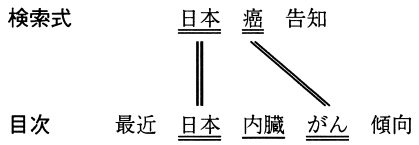
本研究では，このようにして得られた名詞の列を**正規化名詞列**と呼ぶ．

2.4.2 正規化名詞列の柔軟な照合

前述の方法で得られた検索式の正規化名詞列と目次タイトル中の正規化名詞列とを比較して一致した場合，その図書を検索意図に対して重要であるとする．しかし，それらが完全に一致することは少ないと考えられるため，以下のような柔軟な照合を考える．

まず 2.2 節で述べたように検索語のかな拡張を行う．すなわち「癌」と「がん」との照合などを認める．

さらに，検索式の正規化名詞列の連続する一部分が，目次の一部分と照合することと，余分な語が挿入された部分と照合することを認める．たとえば，検索式の正規化名詞列「日本 癌 告知」は，目次の正規化名詞列「最近 日本 内臓 がん 傾向（最近の日本における内臓がんの傾向)」と以下のように照合する．



ここで，検索式の正規化名詞列中の名詞の一致を連続するものだけに制限するのは，連続していない 2 語の照合，たとえば上記の例では「日本 告知」の照合を許すと副作用が大きいと考えたからである．

部分的な一致や余分な語を含む一致は検索意図を正確に反映している可能性が低くなるので，以下のような重みを考える．検索式から得られた正規化名詞列を「$T_1 T_2 \cdots T_n$」，目次タイトル中の一致する正規化名詞列を「$T'_i X_1 T'_{i+1} X_2 \cdots X_j T'_{i+j}$」$(1 \leq i\ ,\ i+j \leq n)$ とする．ただし，T'_i は T_i またはそのかな拡張を表し，X_k $(1 \leq k \leq j)$ は 0 個以上の T'_i $(1 \leq i \leq n)$ を含まない名詞列を表す．

このとき，この照合の重み w_{fm}（fm: flexible match）を以下のように定義する☆☆．

$$w_{fm} = \frac{w_{part} \cdot (j+1) + 1}{w_{insert} \cdot m + 1} \tag{4}$$

ここで m は $X_1, X_2, \cdots, X_j$ に含まれる名詞の総数を表す．w_{part} は部分的な一致に対する重みで，検索式から得られた正規化名詞列中の名詞が，目次タイトル中の名詞とより多く一致する方が重要度が高くなるこ

☆ 名詞の類似度の計算には，NTT の日本語語彙大系[6]の単語の意味属性により体系化された木構造を用いる．NTT の日本語語彙大系は，日本語の単語 30 万語の意味を約 3000 種類の意味属性で体系化したシソーラスである．意味属性体系は日本語の一般名詞，固有名詞の意味的用法がそれぞれ，2751 種，130 種の意味属性で体系化されている．一般名詞意味属性体系は 12 階層の木構造で，固有名詞意味属性体系は 9 階層の木構造からなる．単語 x と y の類似度 sim_{xy} は以下の式で求められる．

$$sim_{xy} = \frac{2L}{l_x + l_y}$$

ここで，l_x ，l_y は x と y の意味属性の木構造の根（Root）からの階層の深さを表し，L は x と y の意味属性で一致している階層の深さを表す．類似度 sim_{xy} は 0 から 1 の値をとる．

☆☆ 式 (4) では，目次の正規化名詞列中の一致した中の余分な語，上記の例では「内臓」にペナルティを与えているが，前後にある余分な語，上記の例では「最近」と「傾向」に対してペナルティを与えていない．これは，前後の余分な語は図書の内容がより限定的であることを示しているが，検索意図を含むという意味では変わらないと考えられるためである．

表 1 単語 A と B の目次の階層構造中での関係
Table 1 Relations between the words A and B in tables of contents.

A = B	A と B が同じタイトル中に存在 ...A.....B...
A > B	A が B より上位のタイトル中に存在 ...A... └──→...B...
A < B	A が B より下位のタイトル中に存在 ...B... └──→...A...
A ‖ B	A と B が同じ階層のタイトル中に存在 ..... ├── ...A... └── ...B...
A # B	上の 4 つ以外の関係でタイトル中に存在

とを表す．また w_{insert} は，目次で一致した正規化名詞列に余分な名詞が挿入されると重要度が低くなることを表す重みである．

たとえば上記の照合の例では，$j = 1$（$T'_1 =$「日本」，$T'_2 =$「がん」），$m = 1$（$X_1 =$「内臓」）であるので，

$$w_{fm} = \frac{2w_{part} + 1}{w_{insert} + 1}$$

となる．

2.5 目次の階層構造

2.3 節で述べたベクトル空間モデルと tf・idf を用いた手法は，目次中に分散している名詞を対象とするため条件のゆるい検索であるといえる．また 2.4 節で述べた正規化名詞列の一致を調べる手法は，1 つのタイトル中での名詞列の現れ方を調べるため，検索意図に対して強い条件の検索となる．これらの中間の位置にある図書検索の方法として，長尾らが提案した目次の階層構造を利用した図書検索の方法がある[1),2)]．

目次のテキストには，章や節の構造による階層構造が存在する．そのため，目次中に 2 つの単語（以下 A と B と表す）が現れる場合，それらの語の間の関係の強さは，階層構造中でのそれらの 2 単語の位置関係により**表 1** のように順序付けできると考えられる．このため A と B という 2 つの単語を検索語とする場合，この表のような 2 単語の関係を持つ図書がその順序で検索意図に対して適当であろうと推測される．

たとえば,「日本」と「宗教」という 2 つの検索語に対して，**図 1** の図書は「日本」>「宗教」という関係があり検索意図に一致すると考えられる．しかし，**図 2** の図書は「日本」#「宗教」という関係であり検索意図に一致しないと考えられる．この場合は表 1 の順序付けを有効に利用することができる．

背教者の系譜
　武田　清子 著
　　まえがき
　　I　**日本**人とキリスト教
　　一　背教者にとっての「壁」
　　二　**日本**文化における「父の像」と「母の像」
　　ー**宗教**的シンボリズムの問題ー
　　三　文化と社会の革新
　　ー「壁」をうがつー
　　.....

図 1 「日本」>「宗教」の関係を持つ目次の例
Fig. 1 An example of “>” relation.

東欧　再生への模索
　小川　和男 著
　　序章　伝統的ヨーロッパへ回帰する東欧
　　第 I 章　もう 1 つのヨーロッパ
　　1　「東欧」は作られた地域概念
　　2　東欧の多様性ー民族・言語・**宗教**・歴史
　　.....
　　第 V 章　東欧経済のこれから
　　1　ユーフォリアから覚めて現実直視へ
　　2　市場化・民営化ペースの再検討
　　3　旧ソ連各国との貿易の見直し
　　4　東欧諸国相互間の経済関係の改善と強化
　　5　**日本**・東欧経済関係と**日本**企業
　　.....

図 2 「日本」#「宗教」の関係を持つ目次の例
Fig. 2 An example of “#” relation.

本研究では，検索式から得られた正規化名詞列中の連続する 2 つの名詞に対して上記の目次階層中の関係を調べる．そしてその階層関係に応じて重み付けされたスコアを与える．重みの計算については次章で詳しく述べる．

3. 本研究の図書検索システム

本研究の図書検索システムは，2 章で述べた方法を組み合わせて，以下の節で示すように 2 段階の検索を行う．

3.1 1 次 検 索

初めに，すべての図書の目次データの形態素解析を行い，すべての名詞を抜き出す．そして，その名詞の wtf・idf を計算し，図書ベクトルを作成する．検索式も同様に形態素解析を行い質問ベクトルを作成する．

図書ベクトルと質問ベクトルの内積の値をその図書の基本点とし，その基本点が 0 より大きくかつ上位 100 位までの図書を **1 次検索結果**とする．次の 2 次検索の処理は，この 1 次検索結果の図書集合のみを対象とする．これは，2 次検索の処理に時間がかかること，また wtf・idf による方法で 100 冊を取り出せば，その中に検索意図に一致する図書のほとんどが含まれていると予想されることによる．

3.2 2 次 検 索

1 次検索結果のそれぞれの図書に対し，以下の処理によって基本点へのスコアの加算を行う．以下の式中で用いられる α_i は，かな拡張された語が一致した場合の重みであり，

$$\alpha_i = \begin{cases} 1.0 & (T'_i = T_i) \\ w_{kana} & (T'_i \neq T_i) \end{cases}$$

で表される．

(1) 正規化名詞列の存在

2.4.2 項で述べたように，検索式から得られた正規化名詞列が目次の正規化名詞列と一致する場合，スコアを大きく加算する．

目次で一致した正規化名詞列が「$T'_i X_1 T'_{i+1} X_2 \cdots X_j T'_{i+j}$」$(1 \leq i\ ,\ i+j \leq n)$ であるとすると，$Score_+$ は次のように計算する．

$$Score_+ = w_{nns} \times w_{fm} \times sum \qquad (5)$$

ここで w_{nns} は正規化名詞列（Normalized Noun String）の一致に対する重みであり，w_{fm} は式 (4) で定めた重みである．また sum は T'_i, $T'_{i+1}, \cdots, T'_{i+j}$ の wtf・idf の総和であり，目次を D として次式のように計算される．

$$sum = \sum_{k=i}^{i+j} \alpha_k \cdot W(D, T'_k)$$

1 冊の目次中の複数のタイトルに一致する場合は，そのすべてのスコアを加算する．

(2) 2 単語の目次階層に 5 種類の関係が存在

検索式から得られた正規化名詞列の中の連続する 2 つの名詞に対して，2.5 節で示した 5 種類の目次の階層関係を調べてスコアを加算する．

検索式から得られた正規化名詞列を「$T_1 T_2 \cdots T_n$」とするとき，その中の連続する 2 単語 T_i，T_{i+1} $(i = 1, 2, \cdots, n-1)$ について，T_i とそのかな拡張，T_{i+1} とそのかな拡張の間の階層関係を調べる．複数の階層関係が存在する場合には，その中で最も強い階層関係をその図書の持つ階層関係とする．各階層関係に対して加算するスコア $Score_+$ は次のように計算する．

$$Score_+ = w_= \times sum \qquad (6)$$

$$Score_+ = w_> \times sum \qquad (7)$$

$$Score_+ = w_< \times sum \qquad (8)$$

$$Score_+ = w_\| \times sum \qquad (9)$$

$$Score_+ = w_\# \times sum \qquad (10)$$

ここで，$w_=$，$w_>$，$w_<$，$w_\|$，$w_\#$ はそれぞれ A=B，A>B，A<B，A‖B，A#B の関係が存在する場合の重みである．また sum は 2 単語 T'_i , T'_{i+1} の wtf・idf の値の和であり，目次を D として次式のように計算される．

$$sum = \sum_{k=i}^{i+1} \alpha_k \cdot W(D, T'_k)$$

1 次検索結果の図書に対してこれらのスコアの加算を行い，得られたスコアの降順に図書を並べなおしたものを **2 次検索結果**とする．

表 2 検索式の諸元
Table 2 Statistics of queries.

検索式総数	202		
名詞総数	602	（1 検索式あたり	2.98）
並列語総数	75	（1 検索式あたり	0.37）
正規化名詞列総数	277	（1 検索式あたり	1.37）

4. 実 験

本研究の実験は，岩波新書 1211 冊の図書の目次を検索対象とした．被験者 26 名にそれぞれ 10 個の検索式とその検索意図を考えてもらい，それらの検索式を用いて検索を行った．1 つ以上の検索結果が得られた 202 個の検索式を以下の実験の対象とした．検索式の諸元は**表 2** のとおりである．

4.1 評価済み図書集合の作成

実験結果を完全な基準により判断するためには，検索意図に一致する図書の集合をすべて保持する必要がある．そのためには，202 個の検索式それぞれに対して 1211 冊の図書のすべてを調べる必要があるが，それは人手による作業が膨大となり困難である．そこで，1 次検索結果，すなわち wtf・idf の基準によってスコアが 0 より大きくかつ上位 100 位までの図書のみを評価対象の図書集合とした．その際，2.3 節で説明した wtf の 3 種類の重み（式 (2)）と w_{kana} は**表 3** の値に設定して検索を行った．

ただし，4.4 節ではかな拡張を考慮する場合と考慮しない場合の比較実験を行うが，1 次検索の結果得られる図書はかな拡張の有無に依存する．そのため，かな拡張を考慮する場合と考慮しない場合の 2 通りの 1 次検索を行い，その結果得られた図書集合の和を評価対象とした．

人手による評価は著者らが行った．評価値は検索意図に対して，満足（3），普通（2），不満（1）の 3 段階とした．これは，満足か不満かという 2 段階評価では判断が困難な場合もあると考えられるためである．

このようにして得られた評価済み図書集合は，1211 冊すべてを調べていないという意味では完全な集合で

表 3 評価済み図書集合作成のための仮の重み
Table 3 Temporary weights for manual evaluation.

	重み
w_{noun}(固有名詞)	2.0
w_{noun}(普通名詞)	1.0
w_{depth}	1.0
$w_{bunsetsu}$	1.0
w_{kana}	0.5

表 4 人手による評価結果
Table 4 Statistics of manual evaluation.

検索式総数	202		
評価対象図書総数	16443	（1 検索あたり	81.4）
評価 3 の図書総数	755	（1 検索あたり	3.7）
評価 2 の図書総数	1008	（1 検索あたり	5.0）
評価 1 の図書総数	14680	（1 検索あたり	72.7）

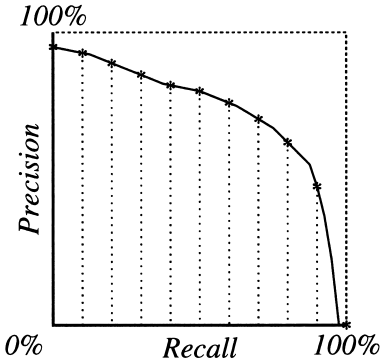


図 3 11 points
Fig. 3 11 points.

はない．そのため Recall は近似的なものとなるが，1 次検索結果の 100 冊の中に検索意図に一致するもののほとんどが含まれていると考えられるので，適当な近似であると考えられる．この人手による評価結果を**表 4** に示す．

4.2 評価基準

情報検索は，Recall と Precision という 2 つの指標により評価されるが，これらは本来トレードオフの関係にあるので，こららを統合した 11 points という評価尺度を用いることが多い．本研究でも，この 11 points を評価尺度として用いる．11 points は以下のように計算する．

(1) 2 次検索結果数を n とする．

(2) 順位 1 から n までの図書を対象に，Recall と Precision を計算する．Recall と Precision は以下のように定義する．

Recall:
$$\frac{\text{順位 1 から } n \text{ の中の正解図書数}}{\text{全正解図書数}} \times 100\ (\%)$$

Precision:
$$\frac{\text{順位 1 から } n \text{ の中の正解図書数}}{n} \times 100\ (\%)$$

全正解図書数は，4.1 節の結果を用いる．

(3) n を 1 つ減らして (2) のステップを $n = 1$ になるまで繰り返す．

(4) 以上のように得られた Recall と Precision の値を Recall を横軸に，Precision を縦軸にしてグラフにプロットして，Recall を 10%間隔で 10 等分して，それぞれに対応する Precision を求めて平均した値が 11 points である（**図 3**）．

202 個の検索式による検索結果から，正解図書が評価 3 のみの場合と評価 3 と 2 の場合のそれぞれの 11 points の値を求め，これらの平均値を本研究の評価基準とする（以下ではこの値を **11ave** と呼ぶ）．

4.3 予備実験

本研究の図書検索システムには，wtf は 3 種類の重み（式 (2)）があり，1 次検索結果へのスコア加算に 9 種類の重み（式 (4)〜(10)）があるが，これらの 11 種類の重みの適切な値を人手で与えることは難しい．そこで最も 11ave が大きくなる重みを決めるため，これらの重みに様々な値を設定して予備実験を行った．

予備実験は次のように wtf の重みの決定とスコア加算の重みの決定の 2 段階に分けた．これは，すべての重みを組み合わせてその値を決定するほうがより適切な値を得られるが，すべてを組み合わせると計算量が膨大となるためである．

(1) wtf・idf の重みの決定方法

すべての目次中の名詞を抜き出し，重みを w_{noun}(固有名詞) $= 0.0 \sim 3.0$, w_{noun}(普通名詞) $= 1.0$, $w_{depth} = 0.0 \sim 5.0$, $w_{bunsetsu} = 0.0 \sim 50.0$ に設定して各図書ごとに wtf・idf を計算する．この wtf・idf によりかな拡張を用いずに 1 次検索結果を抜き出し，それらに 4.1 節で作成した評価済み図書集合を用いて 3 段階評価を与え，11ave を計算した．その結果，11ave が最大となったときの重みは**表 5** のとおりであった．

この結果，$w_{bunsetsu}$は 0.0 となった．すなわち，1 つのタイトル中の文節数は考慮する必要がないということが分かった．

(2) 正規化名詞列に関する重みと目次の階層関係の重みの決定方法

表 5 の重みを用いた wtf・idf により 1 次検索結果を抜き出す．その 1 次検索結果の図

表 5 wtf・idf の重み
Table 5 wtf·idf weighting.

	重み
w_{noun}(固有名詞)	1.7
w_{depth}	1.0
$w_{bunsetsu}$	0.0

表 6 正規化名詞列と目次の階層関係の重み
Table 6 Weights for normalized noun string matching and tables of contents hierarchical structure.

	重み
w_{kana}	0.5
w_{part}	3
w_{insert}	3
w_{nns}	20
$w_{=}$	5
$w_{>}$	4
$w_{<}$	3
$w_{\|}$	2
$w_{\#}$	2

表 7 実験結果
Table 7 Results of experiments.

	かな拡張	正規化名詞列	目次階層	wtf idf	tf idf	11ave
1	○	○	○	○	×	61.8%
2	×	○	○	○	×	59.4%
3	×	×	○	○	×	56.9%
4	×	×	○	×	×	48.2%
5	×	×	×	○	×	52.1%
6	×	×	×	×	○	49.4%

書に対して，かな拡張に対する重み w_{kana} に 0.0 ～ 1.0，柔軟な照合の重み w_{part}，w_{insert} に 0 ～ 5，正規化名詞列が存在する場合の重み w_{nns} と，5 種類の階層関係が存在する場合の重み $w_{=}$，$w_{>}$，$w_{<}$，$w_{\|}$，$w_{\#}$ にそれぞれ 0 ～ 50 の値を設定してスコアを加算し，2 次検索結果を順序付ける．それらに 4.1 節で作成した評価済み図書集合から 3 段階評価を与え，11ave を計算した．その結果，11ave が最大となったときの重みは**表 6** のとおりであった．

4.4 実験と考察

予備実験により決められた重みを用いて図書検索の実験を行った．

本研究の図書検索システムとこれまでの図書検索システムの方法を比較するため，**表 7** のような 6 種類の実験を行った．

この結果から以下のことが分かった．

- 表 7 の比較実験の中で最も検索精度が高かったのは 1 の検索であり，本研究の出発点とした 4 の方法と比較して，13%以上検索精度が上昇した．
- 1 と 2 の検索を比較すると，かな拡張を行う方が検索精度が高いことが分かった．このことから，検索語とまったく意味の同じ語による検索式の拡張であれば検索の精度が上昇することが示された．
- 2，3，4 の検索を比較すると，目次の階層構造の情報だけを用いる検索よりも，wtf・idf や正規化名詞列を組み合わせた検索の方が検索精度が上昇した．これにより，正規化名詞列や wtf・idf や目次階層構造の情報の利用を組み合わせるという本研究で提案する検索方法が有効であることが示された．
- 5 と 6 の検索は，それぞれ wtf・idf と tf・idf を用いて得られた 1 次検索結果を評価した結果である．これらを比較すると，wtf・idf を用いる方が 3%近く検索精度が上昇した．これにより，固有名詞に対する重み付けと目次の階層の深さによる重み付けが有効であることが示された．

表 8 「日本の福祉」の検索結果
Table 8 Retrieval result of “Welfare in Japan”.

スコア	評価	書名	2 次検索での主な要因
11811.6	3	体験ルポ	正規化名詞列
10683.5	3	高齢者医療と福祉	正規化名詞列
3679.9	3	体験ルポ	A=B
1071.2	2	現代日本の民主主義	A>B
1010.1	1	会社本位主義は崩れるか	A>B
723.4	2	現代日本社会と民主主義	A>B
718.2	3	議会	A#B
603.8	3	社会保障	A‖B
475.5	3	GHQ	A‖B
445.1	2	公共事業をどうするか	A#B
445.1	3	障害者は，いま	A‖B
437.7	1	戦後の日本経済	なし
436.1	1	政治改革	A#B
420.9	1	戦後政治史	A#B
372.6	1	国際連合	A‖B
372.6	1	国際連合	A‖B
361.8	1	日本の漁業	なし
328.9	1	日本経済図説	なし
323.8	1	ブナの森を楽しむ	なし
323.7	1	現代を読む	なし

表 7 の 2 の方法による検索で,「日本の福祉」と「遺跡と文化財の保護と保存」という検索式の検索結果でスコアが上位のものを**表 8** と**表 9** に示した．これらは，正規化名詞列を導入したことにより検索意図に合う図書をより上位に得ることができた例である．

5. おわりに

本研究は図書検索をより高精度にするため，目次の階層構造を用いる方法，wtf・idf を用いる方法，自然言語入力を用いる方法，検索式と目次との柔軟な照合

表 9 「遺跡と文化財の保護と保存」の検索結果
Table 9 Retrieval result of “Protection and Preservation of Remains and Cultural Assets”.

スコア	評価	書名	2 次検索での主な要因
31136.9	3	歴史的環境	正規化名詞列
14826.4	3	遺跡保存を考える	A<B
1623.6	3	発掘を科学する	正規化名詞列
1067.4	3	考古学の散歩道	正規化名詞列
602.6	1	世界の自然を守る	なし
493.9	1	アメリカの環境保護運動	なし
451.4	1	プライバシーと高度情報化社会	なし
399.2	2	ナイルに沈む歴史	なし
363.9	1	自然保護という思想	なし
299.3	2	日本人はどこから来たか	なし
287.2	2	日本旧石器時代	なし
153.6	1	世界の公害地図	なし
105.8	2	考古学とは何か	なし
104.1	1	うま味の誕生	なし
103.1	1	検証 日韓会談	なし
82.7	1	環境政策を考える	なし
82.7	1	難民問題とは何か	なし
71.4	1	鼻の話	なし
63.8	1	マルチメディアと著作権	なし
60.5	1	ヴァイオリン	なし

を行う方法を組み合わせた．その中で，正規化名詞列を用いるという検索方法を提案した．この方法は簡単な方法であるが有効であることを示した．

本研究の正規化名詞列は，自然文中の名詞のみを抜き出した名詞列であり，それにより名詞の間の意味的な強いつながりを表現できると考えた．そして正規化名詞列を検索に利用することにより高い検索精度を得られることを示した．

今後さらに検索精度を上昇させるため，より柔軟な照合を行う方法が考えられる．現在のシステムでは，「A B」が「B A」に，また「A B C」が「A C」に一致しない．しかし，たとえば「埋蔵金の発掘」は「発掘された埋蔵金」のような形で現れることもあり，この場合には一致するほうが望ましいといえる．しかし，単純にこれらを一致させるのは副作用が大きいと考えられる．今後は，詳細に意味解析を行うなどすることにより，さらに柔軟な照合を行うことが可能になると考えられる．

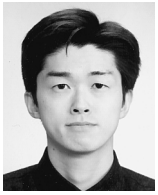
また，検索システム用のシソーラスの作成が必要であると考えている．これまでも報告されたように類義語による検索式の拡張を用いると検索精度が悪化することが多い．しかし本研究の実験により，かな拡張を用いると検索精度が上昇することが確認できた．このため，本研究のようなひらがな形やカタカナ形だけでなく，たとえば「コンピュータ」と「コンピューター」のような表記の揺れや,「計算機」と「コンピュータ」のような意味の同じ語をまとめた検索用のシソーラスが必要であると考えている．このようなシソーラスを大規模に作成し利用することで，検索精度が大幅に向上すると期待できる．

これらのことを追究し，さらに高精度な図書検索システムを開発することが次の目標である．

参 考 文 献

1) 長尾 眞：目次情報などを利用した図書・文献検索方式，情報の科学と技術，Vol.42, No.8, pp.771–718 (1992).
2) 黒橋禎夫，萩原典尚，長尾 眞：目次情報を利用した図書検索システム，情報処理学会情報学基礎研究会，45-5 (1997).
3) 栗山和子：シソーラスを用いた検索式拡張の評価，情報処理学会情報学基礎研究会，52-1 (1998).
4) Voorhees, E.M.: Query Expansion using Lexical-Semantic Relations, *SIGIR ’94*, pp.62–69 (1994).
5) Salton, G. and McGill, M.J.: *Introduction to Modern Information Retrieval,* McGraw-Hill Book Company (1983).
6) 池原，宮崎，白井，横尾，中岩，小倉，大山，林（編）：日本語語彙大系，岩波書店 (1997).

(平成 11 年 3 月 17 日受付)
(平成 12 年 2 月 4 日採録)



白木 伸征

1973 年生．1999 年京都大学大学院工学研究科修士課程修了．現在，（株）豊田中央研究所に勤務．自然言語処理，知識情報処理の研究に従事．

黒橋 禎夫（正会員）

1966 年生．1989 年京都大学工学部電気工学第二学科卒業．1994 年同大学院博士課程修了．博士（工学）．同年，京都大学工学部助手，1998 年京都大学大学院情報学研究科講師，現在に至る．1994 年 4 月より 1 年間 Pennsylvania 大学客員研究員．自然言語処理，知識情報処理の研究に従事．