データベースシステム 118－14
情 報 学 基 礎 54－14
（1999. 5. 17）

WWWページ間の階層構造の推定と検索システムへの応用

原田 昌紀, 佐藤 進也, 風間 一洋
NTT 未来ねっと研究所
東京都武蔵野市緑町 3-9-11

WWWの急速な普及に伴い，サーチエンジンの重要性はますます高まりつつある。しかし，今日のサーチエンジンの出力する膨大な検索結果は未整理で，わかりにくいものとなっている。一因として既存のサーチエンジンは，Webページを独立に扱っており，WWW空間の持つ構造をなんら利用していないことが挙げられる。

本稿では，Webページをまとまりを持ったグループに分ける方法と，サーチエンジンにおけるグループ単位での検索機能について述べる。ロボットで収集したWebページを用いて実験を行ない，本手法の有効性を確認した。

Estimation of Hierarchical Structure of Web for Search Engine

Masanori HARADA, Shin-ya SATO, Kazuhiro KAZAMA
NTT Network Innovation Laboratories
3-9-11 Midori-cho Musashino-shi Tokyo

The rapid spread of World Wide Web lets search engines be more and more important tools. However, their output are chaotic and hard to read. This is because existing search engines treat pages as individuals and do not exploit any structure of the WWW.

In this paper, we describe a method to group WWW pages and the search-by-group feature of our new search engine. Experiments using actual web pages collected by a spider indicate its effectiveness.

1 はじめに

WWWの普及に伴い，サーチエンジンの重要性はますます高まっている。しかし，検索対象となるWebページ数が増大するにつれ，キーワード検索の結果から目的とするWebページを探し出すことが困難になりつつある。そのため，より使いやすい検索結果の提示方法が求められている。

本稿ではWebページをグループ化し，グループ単位で検索を行なう方法について述べる。まず，第2節において従来のサーチエンジンの問題点と既存の研究について述べる。第3節では文書の持つ階層構造と検索システムの関連について述べる。第4節ではWebページをグループ化する手法を示す。第5節では，評価実験について報告する。第6節で課題と今後の展望を述べる。

2 従来のサーチエンジンの問題

2.1 サーチエンジンの検索結果表示

サーチエンジンは，ロボットあるいはスパイダーと呼ばれるソフトウェアを用いてWebページ(多くの場合はHTML形式のテキストファイル)を自動的に収集し，それらのテキストを対象とした全文検索を行なうサービスである。

通常，サーチエンジンは，検索結果をスコア順にランキングして表示する。しかし，実際に利用者が検討できるのは，検索結果の上位数十件程度にすぎない。そのため，検索結果となるWebページ数が大きい場合には，目的とする情報を探すのは困難になる。

さらに，WWWでは複数のWebページをリンクすることによって，一つの主題を表現することが多いにもかかわらず，既存のサーチエンジンはすべてのWebページを独立な文書として検索処理を行ない，常にWebページ単位で検索結果を出力する。一つの主題を表現するように作成されたWebページを，一つの単位として検索処理を行なえば，より使いやすいインタフェースになると思われる。

Excite[6]やLycos[7]など，検索結果をサーバごとにまとめて表示する機能を持つサーチエンジンはすでに存在しているが，より小さくまとまりを持ったページ集合を単位とした検索結果の表示はできない。

また，これらのサーチエンジンの実装は，Webページ単位で検索とランキングを行ない，その上位数十件のWebページを，URLを用いてサーバごとに分類して表示しているにすぎない。これはサーバ単位での検索とは異なる機能である。たとえば，キーワードAとキーワードBの両方を含むWebページを検索し，サーバごとに表示することはできるが，キーワードAを含むページと，キーワードBを含むページの，両方を持つサーバを検索して表示することはできない。

2.2 関連研究

Chenらは，サーチエンジンのユーザインタフェースを改善し，コンテキストの把握を容易にするため，サーバのルートページから検索結果となるWebページへの最短経路を表示するシステムを開発した[2]。しかし，サーバごとにディレクトリ階層が詳細に表示されるため，検索結果数が大きい場合の閲覧性に問題がある。

本稿では，多数のサーバを検索対象とする大規模なサーチエンジンに適した検索結果の提示方法について述べる。

3 文書の階層構造とテキスト検索

3.1 文書の階層構造と検索の単位

一冊の書籍は，章・節・段落というような階層構造を持つ。階層の下に行くほど，特定

の詳細な主題を持つようになり，上に行くほど，全体として大きな主題を持つようになる。オンライン文書もまた，同様の階層構造を持つことが多い。

検索対象となる文書が階層構造を持っている場合，それぞれの階層に対応する単位で検索する機能が考えられる。たとえば，対象が書籍であれば，「○○についての段落を検索する」「○○についての節を検索する」といった具合である。

階層構造を持つ文書を対象とする検索システムは，検索要求の表す主題の大小に応じて，検索処理の単位を選択可能であることが望ましい。検索要求の主題が大きいときに，下位の階層に対応する小さい単位で検索を行なうと，検索結果数が大きくなり，検索結果の一部しか検討できなくなってしまう。逆に，検索要求の主題が小さいときに，上位の階層に対応する大きい単位で検索を行なうと，検索結果数が小さくなると同時に，検索結果あたりの情報量が大きくなることから，個々の検索結果を詳細に検討することが難しくなる。

3.2 WWWの階層構造

WWWはハイパーテキスト・システムであり，リンクによってWebページを様々な順序で閲覧できる。そのため，Webページ間の関係は多様であり，書籍のような明確な階層構造を持たない。

しかし，すべてのWebページが独立に存在しているわけではない。一定の作者によって作成されたWebページ群に着目すれば，複数のWebページがリンクされて一つの主題を表現し，それらがまたリンクされて，より大きい主題を表現するといった階層構造が存在すると考えられる。したがって，サーチエンジンもまた，そのようなWWWの階層構造に対応した検索処理を行なえるべきである。

既存のサーチエンジンの検索機能では，Webページ単位の検索が比較的下位の階層での検索に対応し，サーバ単位での検索結果表示が上位の階層での検索に対応すると考えられる。しかし，その中間の階層での検索機能は実現されていない。

3.3 ランキングと検索の単位

一般に検索システムの利用者は，検索結果をスコアの上位から順に，検索要求に適合しているか検討していく。その際に，本質的に同じ情報を持つ文書が二つ以上現れると，最初の文書が検索目的に合致していたとしても，二つ目以降は実質的な適合度(満足度)が低くなる[1]。

検索処理の単位が小さく，検索要求の主題の大きさに対応していないときも，同様の問題が起きると考えられる。サーチエンジンの場合，同じ文書を構成するWebページが検索結果中に複数現れると，それらのWebページが持つ情報が異なっていたとしても，利用者の満足度を下げる原因になると考えられる。

2.1節で述べた検索結果表示の問題は，サーチエンジンに適切な単位での検索機能が欠けていることが原因であると考えられる。これらの問題は，一つの主題を表現するために作成されたページ集合を単位として検索する機能によって改善されると思われる。

4 Webページのグループ化

4.1 経験則によるグループの識別

サーチエンジンにおいて，3.2節で述べた機能を実現するには，あらかじめ検索対象となるWebページを，各階層に対応する単位にグループ化しておく必要がある。以下では特に，特定のサーバ上のWebページを，一つの主題を表現するために作成されたWebページ集合に分類する方法について述べる。

ここで目的とするのは，異なる作者によって作成されたWebページをテキストの意味内容に基づいてクラスタリングすることではなく，文書の作者が複数のWebページをサーバ上にどのように配置し，リンクしたかを推定することである。

本稿では次の2つの経験則を利用する。

経験則1　同じサーバの同じディレクトリにあるWebページは一定の主題を持った文書を構成する

一つの主題を表現するために複数のWebページを作成した場合，それらを一つのディレクトリの下の配置するのが一般的である。通常，URLのパス部分は，サーバのディレクトリ構成を示しており，ページ集合の境界を知る上で大きなてがかりとなる。

また，まとまりを持つページ集合は，閲覧性を高めるために目次となるWebページを持つと考えられる。

経験則2　一つの文書を構成するページ集合には，その目次となるWebページが存在する。目次となるWebページとは次のような特徴を持つ。

- ページ集合の要素の多くに到達可能なWebページ
- ページ集合に含まれないWebページから多くリンクされているWebページ

たとえば，図1のように`www.yonde.ne.jp`に5つのWebページがあったとする。ここで実線の矢印は同じサーバのWebページからのリンクを，点線の矢印は異なるサーバからのリンクを表わす。これらのWebページに，先に述べた経験則を適用すると，図2のように二つのグループに分割される。一つ目のグループは`index.html`を目次とし，二つ目のグループは`book.html`を目次としている。

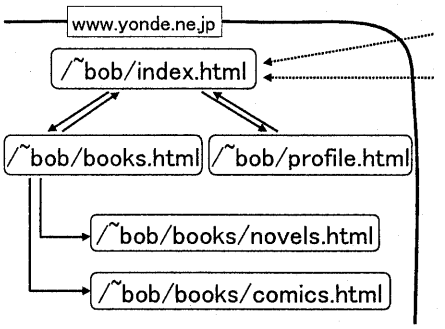

図 1: Webページの例

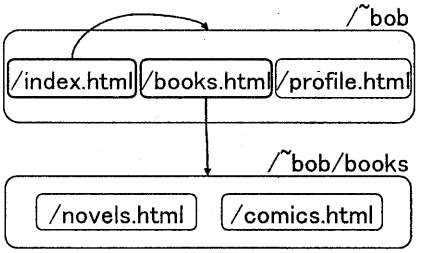

図 2: 図1のグループ化

このようにURLの表記とリンクの参照関係を用いることで，サーバ上のWebページをグループ化することができる。

4.2　グループ化アルゴリズム

前節で述べた方針に基づいて，ページグループを次のように定義する。ページグループは2つの要素からなる。

- ページの集合
- 同じサーバ上にあるインデクスページ

ここで，インデクスページとは4.1節で述べた目次となるページに相当するWebページである。インデクスページはページグループと一対一に対応するが，ページグループ中のページ集合から選ばれるとは限らないとする。

ページグループは基本的にはディレクトリごとに作られ，そのディレクトリに存在する

ページ集合に対応する。ただし，適切なインデクスページが同じディレクトリにない場合のみ，その親ディレクトリに対応するページグループに集約されるとする。仮に図1の例でbooks.htmlがなかったとすると，booksディレクトリはページグループを持たず，すべてのWebページが/~bobディレクトリに対応するページグループに属する。

グループ化アルゴリズムは次の通りである。

1. 収集したWebページをサーバごとに分類する
2. ディレクトリごとに，その上にあるWebページの集合を作成する
3. 深いディレクトリにあるページ集合から順に，対応するインデクスページを求める。適切なインデクスページがあれば，その組をページグループとする。なければ，そのページ集合を，親ディレクトリのページ集合に加える
4. 検索対象となるすべてのサーバ・ディレクトリに対して，手順2,3を適用する

このときインデクスページは次の順序で決定される。

1. ファイル名がindexで始まるWebページ
2. パスが/で終わるWebページ
3. 別のサーバ上のページからもっとも多くリンクされているWebページ
4. 同じサーバ上にあって，集合の要素となるWebページにもっとも多くリンクしているWebページ

最後の条件で，集合の要素となるWebページにリンクしているページが一つも見つからなかった場合は，インデクスページの探索は失敗となる。

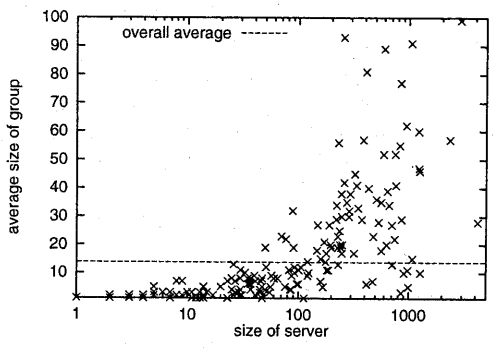

図 3: サーバの大きさとページグループあたりの平均Webページ数

5 評価実験

5.1 Webページのグループ化

4.2節で提案した方法を適用し，Webページをページグループに分ける実験を行なった。検索対象はgo.jpドメインに存在する210ヶ所のサーバから収集したHTMLファイル84,225個である。これらに対応するディレクトリの総数は8,400個で，1ディレクトリあたりのWebページ数は平均10.0個となった。

これらに4.2節で述べた手順を適用したところ，6,101個のページグループに分類された。ページグループあたりの平均Webページ数は，全体としては13.8ページとなり，Webページ数の多いサーバほど大きくなる傾向がみられた(図3)。

5.2 グループ単位検索の実装

開発中の全文検索システムに，ページグループ単位での検索機能を実装するために，通常のWebページ単位の全文検索用の索引に加えて，ページのIDから，そのWebペー

ジの属するページグループのインデクスページのIDを得るための表を用いた。

検索時にはまずページ単位でキーワード検索とスコアの計算を行なう。スコアは，単一のキーワードで検索を行なった場合にはtf·idf法に基づいた方法で計算する [4]。

$$nfreq_{ij} = \log(freq_{ij} + 1) / \log(length_j + L)$$
$$IDF_i = \log(N/n_i)$$
$$weight_{ij} = nfreq_{ij} \times IDF_i$$

ここで，$freq_{ij}$ は文書 j 中にキーワード i が出現した回数，N は全文書数，n_i はキーワード i が出現した文書数，$length_j$ は文書 j のサイズ (文字数) である。極端にサイズが小さい文書のスコアが大きくなりすぎるのを抑制するため，定数 L で補正してある。L は平均文書サイズの 1/4 とした。

続いて，表を参照して各Webページに対応するページグループを求める。ページグループのスコアは，それが含むWebページのスコアの最大値とし，最後にそのスコア順でページグループをソートして表示する。

利用者には，各グループのインデクスページと，グループ中で実際に検索結果となったWebページの情報が表示される (図4)。

5.3 有効性の評価

5.1節でグループ化したWebページを対象として，120種類のキーワードで検索を行なった結果を表1に示す。キーワードは無作為に決めたが，10ページ以上の検索結果が得られるように選んでいる。

表 1: 実験に用いたキーワードと検索結果数 (抜粋)

キーワード	ページ数	グループ数
環境問題	995	394
情報公開	598	262
ダイオキシン	348	166
クローン	243	120
介護保険	208	93
地熱	120	57
消費者物価指数	84	25
雇用機会均等法	31	18
ワシントン条約	13	8

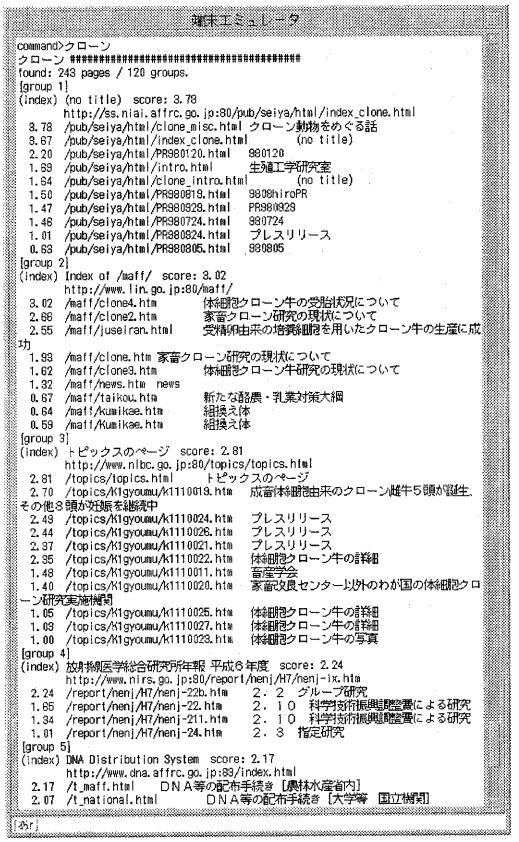

図 4: ページグループ単位の検索結果表示

検索結果数の平均は324ページ，123グループであり，グループ化によって検索結果リストの長さが平均0.38倍に短縮される効果があった。また，検索結果となったそれぞれのグループのヒット数 (ページグループ中でキーワードを含むWebページの数) は検索結果Webページの総数にかかわらず，ほぼ一定になる傾向がみられた (図5)。これらのことから，少なくとも検索結果数が10件以上のキーワードについては，検索結果リストを短縮する効果があるといえる。

一方，それぞれのキーワードでの検索結果

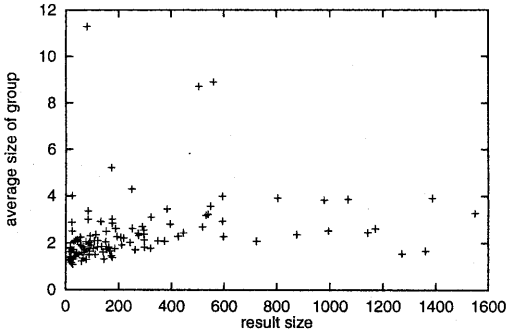

図 5: 検索結果の総ページ数とグループあたりの平均ヒット数

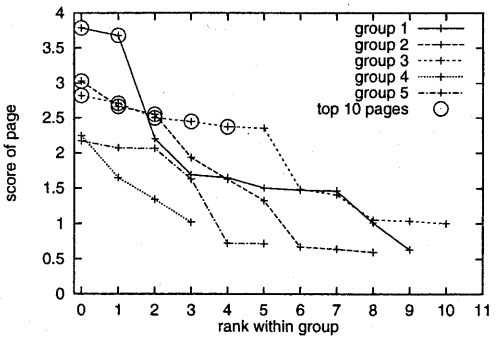

図 7: 「クローン」の検索結果上位 5 グループ中のスコア分布

をみると，ヒット数の大きいグループが，ランキングの上位となる傾向がみられた。図6は，キーワード「クローン」で検索を行なったときの，グループのヒット数とランキングの関連を表している。

ヒット数の大きいグループは，ページグループとしての適合度が高いと考えられる。サーチエンジンの利用者はグループをスコア順に検討すると予想されるため，そのようなグループが上位にランキングされるのは望ましい性質であるといえる。

5.2節で述べたように，グループのスコアは，グループ中で検索結果となったページの

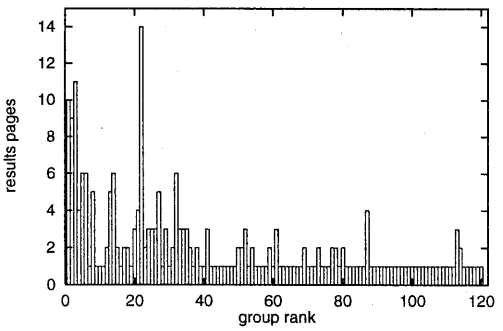

図 6: 「クローン」の検索結果となったグループのヒット数 (スコア順)

スコアの最大値を用いて計算されている。したがって，ヒット数の大きいグループほど上位にランキングされる確率が高くなるのは当然ともいえるが，それだけではない。たとえば「クローン」の検索結果の上位 10 ページは上位 3 ページグループに属しており，スコアの高い Web ページが少数のグループに属する傾向がみられる (図7)。すなわち，利用者は少数のスコアの高いグループを検討するだけで，多くのスコアが高い Web ページを検討できるようになると期待される。

以上の結果から，グループ単位で検索結果を表示することによって，利用者の負荷が軽減できると結論できる。また，本稿で示したグループ化手法が，妥当性を持っていることが示唆される。

6 まとめ

本稿では Web ページをグループ化し，グループ単位で検索する手法について述べた。評価実験によって，グループ化を行うことで，検索結果の閲覧効率が高くなることが確かめられた。

提案したグループ化方法の妥当性のさらなる検証は今後の課題である。本手法は URL の表記を特に重視しているが，リンクによる

参照関係の解析をより重視した方法も考えられる。たとえば，本手法ではインデクスページからページグループ中の全Webページへ到達可能とは限らないという問題がある。

現状ではWebページ間の関係は，リンクなどから推定するしかない。しかし，W3Cで制定中のRDF[5]というメタデータの枠組みを用いれば，Webページ間の関係も明示的に記述することができるようになる。普及には時間がかかることが予想されるが，RDFの利用も検討の余地がある。

今後は，現在開発中のサーチエンジンを一般に公開し，アクセス履歴を調べることで，サーバ単位・グループ単位の検索の実用性を検証する予定である。

参考文献

[1] Robert R. Korfhage: "Information Storage and Retrieval" Wiley Computer Publishing, 1997.

[2] Michael Chen, Marti A. Hearst: "Presenting Web site search results in context: a demonstration" In *SIGIR'98: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp.381. August 1998.

[3] Michael Chen, Marti A. Hearst: "Cha-Cha: Contextualizing Intranet Search Results" `http://cha-cha.berkeley.edu/`

[4] Donna Harman: "Ranking Algorithms" In William B.Frakes, Ricardo Baeza-Yates, editors: *Information Retrieval: Data Structures & Algorithms*, pp.363–392, Prentice Hall, 1992.

[5] "Resource Description Framework (RDF)" `http://www.w3.org/RDF/`

[6] "Excite" `http://www.excite.com/`

[7] "LYCOS Home Page" `http://www.lycos.com/`