Webコミュニティの知識に基づく情報検索手法の評価

久我昌崇[†] 中所武司[††]

近年インターネットの普及により，Web 上の情報量は増加の一途を辿っている．そして情報量の増加に伴い，情報検索の質の低下という問題が起っている．すなわち，検索対象が大きくなるためにユーザの要求を満す結果を得るのがますます困難になってきている．こういった適合率の低下問題の解決案としてリンク分析型の情報検索システムが注目されており，Google,Cleverproject 等が知られている．しかしこれらにはそれぞれ，検索キーワードの扱いに柔軟性がない，検索時間が長いといった問題点がある．そこで，我々はこれらの問題点を解決し，適合率の高いサイトを返すことのできる新たなリンク分析型の情報検索システムを提案する．この方式は，直接検索キーワードにマッチする文書を検索するのではなく，まずキーワードに関連する Web 上のコミュニティを求める．そしてそのコミュニティの知識を利用して結果を求める．このアルゴリズムにより，同義語検索を可能にし Google の柔軟性の欠如問題を解決した．また，Cleverproject よりも少ない処理で検索を行なうことができる．本論文ではこの Web コミュニティ方式の概要を述べ，試作したプロトタイプシステムの結果を考察する．また，既存のシステムとの比較を行ないその問題点を解決できることを示す．

The Evaluation of the Information-Retrieval Method Based on Web-Community's Knowledge

Masataka Kuga[†] and Takeshi Chusho[††]

In these days, as Internet grows, information on the Web has been increasing. It caused a problem of relevancy drop on Information Retrieval(IR). Due to the excess of search result, it becomes harder for users to find the result which satisfy their request. Link Analysis methods which solve this matter, have been watched with keen interest. For example, there are Google and Cleverproject. But these systems also have problems. Google has less flexibility on handling keywords to search, and Cleverproject takes too much time on searching. Therefore, we suggest a new IR method based on Link Analysis type which solves these problems and marks high relevancy. This method doesn't search the sites which match the keywords to search but gets the Web-Community relevant to the keyword, and gets the result based on its knowledge. By this algorithm, our system solved the Google's problem, and is able to search a synonym. Furthermore, this system is able to search with less process than Cleverproject. In this paper, we refer the outline of Web-Community, and consider the result of the prototype of that system, and compare it with the exsisting systems to indicate that the system can solve the problems.

1. はじめに

ネットワークの普及により多くの人々が Web を利用するようになり，その情報量が爆発的に増加し続けている．例えば，Netcraft Web Surver Survey[1)] の調査によると 1999 年 2 月の時点で全世界のサーバ数はおよそ 430 万であり，2000 年 11 月の時点では 2300 万強である（**図 1** 参照）．つまり，わずか 21ヶ月で約 5.5 倍になっている．一方，'99 年 2 月の時点の全世界の Web サイト数は，NEC 北米研究所の Steve Lawrence 等[2)] の研究によると 8 億強である．そこで，サーバ数に比例して Web サイト数が増加すると仮定すると，2000 年 11 月の時点での全世界の Web サイト数は 44 億程であることが予想される．

このように Web 上の情報量が著しく増加することによって情報検索の分野にも大きな問題が起きている．一つはサイトの収集が Web の増加に追い付かないという量の問題である．そしてもう一つは，情報量の増加に伴い検索結果の適合率が低下しているという質の問題である．つまり，検索結果の量が増加しているた

† 明治大学大学院理工学研究科基礎理工学専攻情報科学系
Computer Science Cource, Major in Sciences, Graduate School of Science and Technology, Meiji University.
†† 明治大学理工学部情報科学科
Department of Computer Science, Science and Technology, Meiji University.

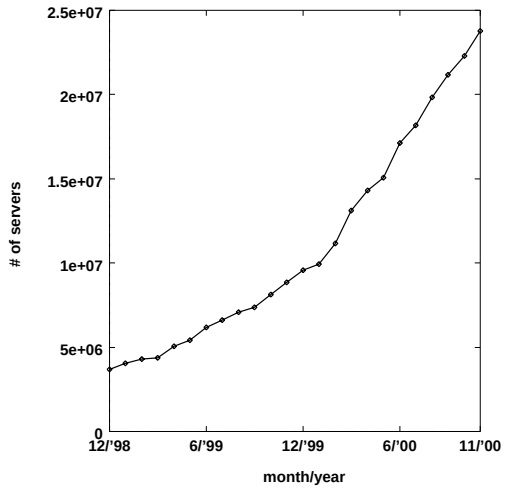

図 1 世界のサーバ数
Fig. 1 The number of servers in the world by Netcraft Web Server Survey

めに，ユーザが求める情報を得るのが困難になっているのである．本論文ではこの検索結果の適合率の問題に焦点を当てる．

こういった適合率の問題は，従来のテキスト分析型システムがサイトの質を適切に判断していないことに由来する．テキスト分析型では，サイトに含まれるキーワードのフォントの種類や大きさ・位置などからサイトのランキングを行なっているが，サイトの質の判断は不充分である．例えば，多くの検索システムはHTML の TITLE タグにキーワードが含まれているサイトのランクを高くするが，これはサイトの質には全く関係がない．このように，テキスト分析型ではサイトの質を適切に考慮してランキングを行なっていないので，情報量が増加するとそれに伴い結果が多くなり適合率が低下してしまう．だが，サイトの質を考慮してランキングを行なえば，ユーザは有用なサイトから順に見ていくことができるので，情報量の増加問題に対処することができる．

そういった背景から，リンク情報を用いてサイトのランキングを行なうリンク分析型の情報検索システムが注目を集めている．Google[3)4)5)] や Cleverproject[6)] などがそうである．これらのシステムは Web のハイパーリンク構造を分析することで，サイトの有用度の評価を行なっている．被リンク（backward-link）の多いサイトほど有用であるという基本概念に基づいてサイトのランクを計算することで，一般的に有用であると思われるサイトを判別することを可能にした．これらのリンク分析型のシステムは従来のテキスト分析型よりも高い適合率を発揮するので，Web 情報検索に大きく貢献した．

しかし，これらのシステムにもそれぞれ問題点ある．Google は検索キーワードに完全一致する文書しか結果として返さないため，検索に柔軟性がない．また，Cleverproject は検索処理が多いため検索時間がかかるという欠点を持つ．

そこで，我々は Web コミュニティ方式[7)] で Google や Clever とは異なるアルゴリズムを用い，この問題の解決を図った．本方式は従来の検索システムとは異なり，検索キーワードから検索結果を直接求めることはしない．まず，キーワードに関連する Web 上のコミュニティを求め，その知識を利用して検索を行なう．これにより，Google のキーワード偏重による柔軟性の欠如問題を解決することができる．また，Cleverproject もコミュニティに似通った概念のグループを求めることで検索を行なうのだが，本方式ではより精選されたコミュニティをより単純な方法で求めるので，少ない処理で優れた結果を返すことを可能にしている．そして，Web のハイパーリンク情報を利用してサイトの有用度判断を行なっているので，有用度の高いサイトを結果として返すことができる．すなわち，サイトのランキングを適切に行なうことで情報量の増加に伴う適合率の低下問題を解消している．

本論文の構成は以下の通りである．２章では，既存のテキスト分析型情報検索システムとその問題点についてそれぞれディレクトリ型とロボット型とに分けて述べる．３章では，Google，Cleverproject を取上げ，既存のリンク分析型情報検索システムとその問題点について述べる．そして，４章で Web コミュニティ方式について述べる．それから５章で，本方式の検索結果について考察する．６章では既存のシステムとの比較評価を行なう．最後に７章でまとめる．

2. 既存のテキスト分析型情報検索システム

従来型のテキスト分析型情報検索システムは大きく分けてディレクトリ型とロボット型の２種類があり，それぞれ長短所が異なる．

2.1 ディレクトリ型検索システム

ディレクトリ型とは Yahoo[8)] に代表される手動でサイトを登録する検索システムである．これには一般的に次のような特徴がある．

- 人手で登録しているのでサイトの質が高い
- 人手で登録するためコストが高い

しかし，情報量の増加に伴い登録コストが増しているので，サイトが適切なディレクトリ下にない，もしくはサイトの質が低いなどの問題点が大きくなってきている．

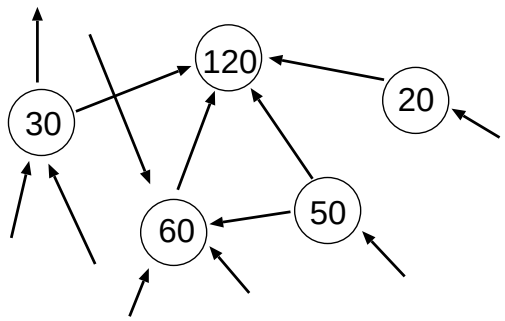

図 2 Google のサイト評価アルゴリズム
Fig. 2 PageRank algorithm on Google

2.2 ロボット型検索システム

ロボット型は goo[9] 等に代表されるシステムで，web ロボットと呼ばれるソフトウェアによりサイト収集を自動的に行なう．これには次のような特徴がある．

- サイト収集を自動化しているのでコストが小さい
- 大量のサイトを収集することができる
- spam サイトや有用度の低いサイトも一様に収集してしまうので適合率が低い

このようにロボット型は大量のサイト収集を可能にする一方，適合率が低いことが問題となっている．Web の情報量は増加し続けているので，これでは検索結果の適合率は益々下がっていくことになる．

3. 既存のリンク分析型情報検索システム

リンク情報をサイトランクに用いて適合率の改善を計ったリンク分析型情報検索システムの代表である Google と Cleverproject を取上げる．

3.1 Google

Google は Stanford University の Sergey Brin,Lawrence Page らによって開発されてシステムで現在は商用化されている．このシステムの最大の特徴はリンクを他のサイトへの推薦とみなしている点である．Google が用いている PageRank と呼ばれるサイトランキング方法は**図 2** のようになっている[10]．

つまり，多くの有用なサイトにリンクされているほどそのサイトのランクが高くなるようになっている．また，リンク先のサイトに与えるランクはリンク数で割ったものになっている．例えば，図 2 のランク 50 のサイトはリンクを 2 つ持っているので，リンク先サイトそれぞれに 25 ずつのランクを与えている．

このシステムの特徴は次のようになっている．

- リンク分析をサイト評価に利用しているので検索結果の適合率が高い
- キーワードがサイト内の文書もしくはそのサイトを指すアンカーに完全一致する場合のみ，そのサイトを結果として返す
- キーワードの出現位置やフォントの種類等のテキスト情報もサイトランクに考慮している
- 検索結果のリンク先を調べ，そのリンク先サイトに検索キーワードがある場合，結果のサイトのランクを高くする

一方，問題点としては次のようなものがある．

- キーワード完全一致の文書しか返さないためキーワードの扱いに柔軟性がない

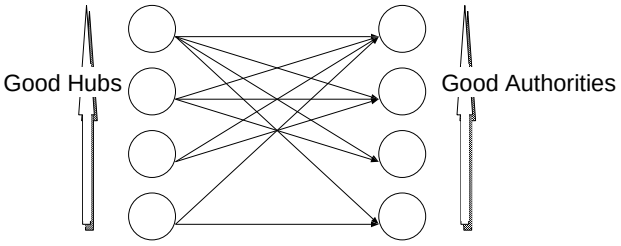

図 3 Hub と Authority の概念図
Fig. 3 Hubs and Authorities

3.2 Cleverproject

Cleverproject は IBM Almaden Research Center で開発されたもので，Google とは異なりサイトを Hub と Authority という 2 つの概念に分けている[11)12)]．優れた Authority にリンクしているサイトが優れた Hub であり，優れた Authority とは優れた Hub にリンクされるものである，という相互再帰的な計算によりそれぞれのスコアを求める（**図 3**）．

例えば，図 3 では左列（Hub）の上から 3 番目と 4 番目のサイトは共に 2 つのリンクを持つが，リンク先サイトのランクがより高い 3 番目のサイトの方が高いランクになる．また，右列（Authority）の上から 3 番目と 4 番目のサイトはともに 2 つのサイトからリンクされているが，被リンク先サイトのランクがより高い 3 番目のサイトの方が高いランクになっている．

このシステムのアルゴリズムは次の様になっている．

(1) 既存の検索システムである Altavista[13] 等に検索キーワードを送り，結果を得る
(2) その結果のサイト群にリンクしている／されているサイトを求め，それに最初の結果を足したものをルートセットとする
(3) ルートセット内の参照関係を分析し，Hub ポイント，Authority ポイントをそれぞれ求める
(4) ポイントの高い順に結果として返す

このシステムの特徴は次のようになっている

- リンク分析によりサイト評価を行なっているので検索結果の適合率が高い
- リンク情報を元に結果を求めているので，検索

キーワードを含まないサイトを検索することができる

一方，このシステムは上記のアルゴリズムからわかるようにルートセットを求めさらにその参照関係を計算しなければならないので，検索処理時間が長いという問題点を抱える．

4. Web コミュニティ方式の概要

4.1 基 本 概 念

本稿で提案する Web コミュニティ型検索システムは，知らないことは知っている人に聞く，という情報を探す際の基本的な考えを礎にしている．Web には，ユーザが求める情報に関する知識を持つ人が大勢いる．そこで，この “知っている人” の知識を検索に応用しようというアプローチである．そして，あるトピックに詳しい “知っている人” が作成したサイトであるリンク集の利用を提案する．リンク集とはあるトピックに関するリンクを多数持つサイトであり，そのトピックに詳しい人が作成しているのでリンク先サイトの質が高いことが予想される．よって，このリンク集の作者を “知っている人” とみなし，その知識をリンクとみなす．つまり，リンク集を得ることができれば，そのリンク先は質の高いサイトであることが予想されるので，サイトのランキングに応用できるのではないかと考えた．

また，リンク集は一つだけでは意味をなさないのでその集合を得る必要がある．そして，リンク集の集合，すなわちあるトピックに関する専門家の集団を我々は Web コミュニティと名付けた．

尚，文献[14)15)] でも共通のトピックを持つ Web 上のコミュニティについて論じられているが，これは Web の構造分析を主眼にしたものであり，情報検索に用いる本稿とは定義やアプローチが異なる．また，文献[16)] でもリンクを多数持つリンク集のようなサイトはユーザにとって有用であると述べている．

4.2 特　　徴

このシステムの特徴は次の通りである．

- 人手で作成された質の高いリンクを持つリンク集をコミュニティとして利用しているので検索結果の質が高い
- キーワードの表記の違いにとらわれず柔軟性のある同義語検索を行なうことができる
- Cleverproject よりも少ない処理数で検索を行なうことができる

4.3 アルゴリズム

このシステムのアルゴリズムは次のようになっている（**図 4** 参照）．

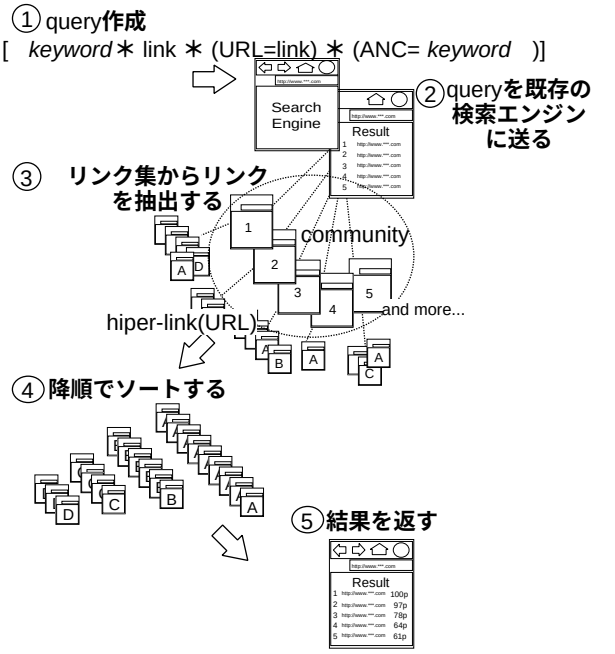

図 4 web コミュニティによる検索アルゴリズム図
Fig. 4 the algorithm of Web Community search

(1) リンク集を求めるために入力キーワードを変形した query を作成する
(2) query を既存の検索エンジンに送り，リンク集の集合を得る
(3) そのリンク集の集合からリンク先の URL を抽出する
(4) その URL 群を降順でソートする
(5) 検索結果として返す

ここで図 4 の query = [入力 *keyword* * link * (URL=link) * (ANC=入力 *keyword*)] とはリンク集を得るために入力キーワードを変形したものである．入力キーワードと文字列 link をテキストに含み，URL に文字列 link を含み，ANC(アンカー) に入力キーワードを含む，という制限によりリンク集を得る．

尚，既存の検索エンジンは，URL 情報とアンカー情報を指定することができる infonavigator[17)] を用いている．

また，この query で得られる集合のうち，入力キーワードに関する複数のリンクを持つものの割合は，結果上位 100 件を調査した所 90 %であった．そして，同様にキーワードに関連するサイトへ 1 つ以上のリンクを持つサイトの割合は 100 %であった．この数値は改善の余地はあるものの，非常に高い精度でリンク集の集合を得ることができることを示している．

	100	200	300	400
平均リンク数	49.41	33.59	38.02	41.22
サイト数合計	3942	4808	7795	10811
平均被リンク数	1.41	1.40	1.46	1.53
上位 20 件平均	15.45	18.15	26.25	31.80
その他平均	1.18	1.33	1.40	1.47
割合（%）	92.24	93.47	90.71	90.64

表 1 各コミュニティ規模におけるリンク数データ
Table 1 the data at each size of Community

5. 検索結果

5.1 結　　果

Web コミュニティ方式のプロトタイプを作成し，実際に検索を行なった．まず，コミュニティの規模を 100,200,300,400 と変化させた場合のデータを**表 1** に表す．

ここで，平均リンク数はコミュニティを構成する各リンク集の持つ平均リンク数を表す．そして，サイト数合計とはコミュニティからリンクされている結果サイトの総数である．それから，平均被リンク数とはそれらの結果サイトそれぞれがリンクされている数の平均である．また，結果上位 20 件の被リンク数平均とその他大半のサイトの被リンク数平均も表してある．最後の割合とは，被リンク数が 1 ないし 2 のサイトの全体に占める割合である．

この表 1 で興味深いのは，まず平均リンク数である．米 Cyveillance 社の調査[18] によると Web サイトの平均リンク数は 28.6 なので，Web コミュニティのそれは大きく上回ることになる．これはあるトピックに関するリンクを多数持つリンク集の特性を表していると言える．また，それに関連してサイト数合計も通常想定される値よりも非常に高くなっている．

次に被リンク数だが，上位 20 件はコミュニティの規模が大きくなるにつれ多くなっているものの，全体平均とその他の平均はほとんど変化していない．そして，被リンク数が 1 ないし 2 のサイトの割合がいずれも 90 %を越えている．これらのデータから，結果サイトはコミュニティの意見が共通するごく一部の部分と意見が分散するその他大半に二分化されていると考えることができる．

そこで，この仮説を検証するために，各コミュニティ規模における被リンク数とサイト数の関係をグラフに表した（**図 5** 参照）．

これからわかるように，コミュニティの規模に関わらずヒストグラムはほぼ同じ形をしている．よって，結果サイトが一部の高評価もしくは著名なサイトと，

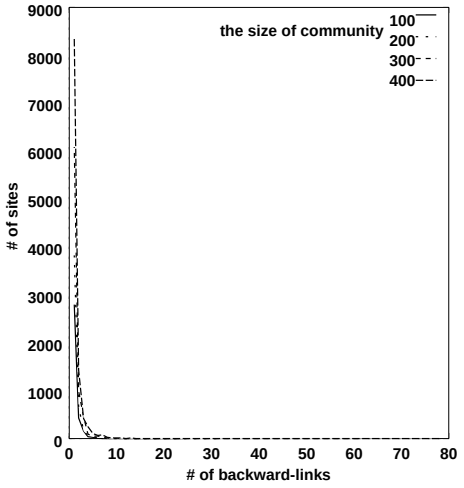

図 5 被リンク数とサイト数の関係
Fig. 5 Histogram of backward-link counts

	URL	Pts.
＊	http://www.yahoo.co.jp/	31
＊	http://www.forest.impress.co.jp/	29
	http://java.sun.com/	26
＊	http://www.vector.co.jp/	25
	http://java-house.etl.go.jp/ml/	24
＊	http://www.goo.ne.jp/	21
	http://www.webcity.co.jp/info/andoh/java/javafaq.html	18
	http://www.sun.co.jp/java/	18
	http://www.javasoft.com/	18
	http://www.java-conf.gr.jp/	18

＊：結果として不適切なサイト
表 2 本方式の初期結果 (keyword=java)
Table 2 the early result of Web-community search system (keyword=java)

その他多くの低評価もしくは無名なサイトとで構成されていることが示された．尚，これは文献[19] によると Web 全体の傾向である．そして，本方式はこのごく一部の高評価もしくは著名なサイトを結果として返すことを目的とする．

また，コミュニティの規模だが，結果として返したい有用なサイトはごく一部であることから，規模を大きくしても結果は変わらないことが予想される．そして規模を大きくすると計算量も多くなるので，規模は小さい方が好ましい．そこで今回は経験則より 200 件が適当と判断し，この前提の元に話を進めていく．

次に，具体的な検索結果の上位 10 件を**表 2** に示す．尚，キーワードは [java]，コミュニティの規模は 200 件である．

これからわかるように，この結果には java という検索キーワードに不適切なサイトが含まれる．すなわち，先のアルゴリズムを適用しただけでは有用な結果を得ることはできない．そこで，この検索結果の改善を 5.2 節で行なう．

5.2 結果の問題点と改善法

初期結果には次のような不適切なサイトが存在するので，それを改善する．

- 検索キーワードと関係のない著名サイト
  表 2 の 1 位の www.yahoo.co.jp や 2 位の www.forest.impress.co.jp などは検索キーワードに関係なく一般的に著名なサイトである．文献[20]によると Web では著名サイト程多くのリンクを集める傾向があるので，被リンク数でカウントするとこういったサイトが上位に来てしまう．これらは検索結果として不適切なので，取り除かねばならない．具体的には，多くのキーワードの結果上位に共通して現れるものを予め登録しておき，それに適合したものを排除する手法を取る．こういった検索キーワードによらない著名サイトは，我々の経験則によると日本語ドメインに限れば 20 種類程である．
- URL 表記が異なるが同じ内容のサイト
  コミュニティのサイト作成者それぞれで同じ内容を指すにもかかわらず URL 表記が異なる場合がある．具体的には，“/” や “index.html” そして “www.” の有無である．こういった違いを吸収して，リンク数をカウントする必要がある．例えば，表 2 の 3 位の java.sun.com/（26Pts.）には同じ内容を表すサイトとして “/” のない java.sun.com（1Pts.）や “www.” のある www.java.sun.com/（4Pts.）があるので，これらを合計して 31Pts. として返す．また，同様に “index.html” の有無だけが異なるサイトも同一であるとみなし，Pts. を合計する．
- ドメイン名が異なるが同じ内容のサイト
  ミラーサイトの存在により，ドメイン名が異なるにもかかわらず同じ内容を指すサイトがある．例えば，表 2 の 7 位の www.webcity.co.jp/info/andoh/java/javafaq.html（18Pts.）と tech.webcity.ne.jp/~ andoh/java/javafaq.html（5Pts.）である．こういったサイトが同一であるか URL を用いて判断する．具体的には，ドメイン名が異なり，その下のディレクトリ階層が 3 つ以上同じ場合はそれらのサイトを同一であるとみなす．よって，この場合は条件を満す（“~ ” は吸収）ので合計 23Pts. として上位のサイトが下位サイトを吸収する．この手法は実験の経験則から得たものである．また，文献[14)21)]でも URL 情報を用いて，本方式とは異なる手法でミラーサイトの判別を行なっている．

URL	Pts.
http://java.sun.com	31
http://java-house.etl.go.jp/ml/	24
http://www.webcity.co.jp/info/andoh/java/javafaq.html	23
http://www.webcity.co.jp/info/andoh/java/javanew.html	21
http://www.sun.co.jp/java/	19
http://www.javasoft.com/	19
http://www.java-conf.gr.jp/	19
http://www.sun.co.jp/	17
http://www.javacats.com/JP/	16
http://www.dmz.hitechi-sk.co.jp/Java/Tech/	15

表 3 本方式の改善後結果 (keyword=java の場合)
Table 3 the result after the improvement of Web-community search system (keyword=java)

この改善処理を行なった後の本方式の結果上位 10 件を**表 3** に示す．

この改善結果と表 2 を比較すると，著名サイトが排除されたことで 10 件のうち 4 件が新しくなっている．また，改善後 1 位の java.sun.com は “www.” や “/” の有無だけが異なるサイトのポイントが足されている．そして，改善後 3 位の www.webcity.co.jp/info/andoh/java/javafaq.html は先に述べたようにミラーサイトが存在し，そのポイントを足したのでランキングが上昇している．

このように，結果を改善することでより適切にランキングを行なうことを可能にした．

6. 既存のリンク分析型システムとの比較

6.1 Google との比較

Google のアルゴリズムは，テキストかそのサイトへのアンカーに入力キーワードを含むサイトしか結果として返さない．そのため適合率は高くなるが，検索キーワードの扱いに柔軟性がない．しかし，本システムでは同義語にも対応する柔軟性のある検索を行なうことができる．

一例として，キーワード [サッカー] で検索するケースを考える．この場合，Google の結果 1 位には www.jfa.or.jp/というサイトが返される．しかし，同じ意味を表すキーワード [soccer] で検索するとこのサイトは上位 100 件にも返されない．この理由は Google がキーワードを偏重していることによる．つまり，サイト www.jfa.or.jp/はサッカーという単語は含むが soccer という単語は含まないので，このサイトを指すアンカーに soccer という単語がない限り結果として返されない．このように，Google は自身が重要であるとみなしているサイトであるにもかかわらず，キーワードに完全一致することがないと結果として返すことができない．これではユーザは同義語についても調べな

	URL	Pts.
＊	http://www.nikkansorts.com/	39
＊	http://www.jfa.or.jp/	28
＊	http://nakata.net/	27
	http://www.fifa.com/index.html	24
	http://sports.yahoo.co.jp/soccer/2002c/	18
＊	http://www.j-league.or.jp/	17
	http://www1e.mesh.ne.jp/soccer/	17
	http://www.j-ole.com/	16
＊	http://www.yomiuri.co.jp/hochi/soccer/index.htm	16
＊	http://www2s.biglobe.ne.jp/ s-masaru/	16

＊:キーワード soccer を含まないサイト

表 4 本方式の結果 (keyword=soccer の場合)

Table 4 the result of Web-community search system (keyword=soccer)

い限り，良い検索結果を得ることができない．また，このような問題に対する従来の解決法としてシソーラスを用いた同義語検索があるが，これには OR 検索を行なうと検索結果が大量になり適合率が低下するという問題点がある．

こういった問題はキーワードに重きを置きすぎていることに由来する．それに対して，本方式では入力キーワードはコミュニティを求める際にのみ用い，そこから結果を求める際にはリンク情報を利用する．つまり，検索結果はコミュニティのリンク情報によるのであって，入力キーワードは直接的には関係しない．よって，キーワードに関連するがキーワードを含まないサイトを求めることができるので，結果的に同義語検索を行なうことができる．そして，リンク集という精選された質の高いリンクを持つサイト群を利用しているので適合率も高い．

例えば，**表 4** はキーワードが soccer の場合の本方式の結果を示したものだが，soccer という単語を含まない www.jfa.or.jp/を結果として返している．また，＊印の付いているその他 5 件も同様に soccer という単語を含まない．

このように Web コミュニティ方式では，Google の検索キーワード偏重問題を解決することができる．

6.2 Cleverproject との比較

6.2.1 検索処理の比較

Cleverproject との検索処理の比較を行なう．本システムと Cleverproject との検索処理は主に Web アクセスと HTML 解析の 2 つである．この 2 つの処理をそれぞれどの位行なうのか**表 5** に記した．尚，本方式のコミュニティの規模は 200 件である．

これからわかるように，本方式の方が処理数が少ない．この理由としては検索対象となるグループの求め方が異なることが挙げられる．3.2 節で触れたように，Cleverproject はルートセットを求める際に検索キー

	Cleverproject	本方式
Web アクセス	401	201
HTML 解析	401	201
計算方法	相互再帰計算	降順ソート

表 5 Cleverproject と Web コミュニティ方式の検索処理比較表

Table 5 the comparative table of the search process steps of Cleverproject and Web-community search

ワードをそのまま用いて結果を得る．そして，その結果のリンク／被リンクサイトを求めるので Web アクセスを大量に行なわなければならない（リンク先 URL を求めるのに 200 回，被リンク先 URL を求めるのに 200 回）．一方，本方式では検索キーワードを変形して独自の query を作成することにより，コミュニティの URL を 1 度の Web アクセスで求めている．そして，それに加えてコミュニティの規模数だけそれぞれの処理を行なう必要があるが，5.1 節で述べたようにコミュニティの規模は小さく設定することができるので 200 回ずつで済む．

また，最後に検索結果を求める際に Cleverproject は相互再帰計算を行なうのに対して，本方式では単純に降順ソートを行なう．この点でも本方式の方が少ない処理数で検索を行なうことができる．

これらの違いにより，本方式では Cleverproject よりも少ない処理で検索を可能にしている．

6.2.2 検索結果の比較

Cleverproject と本方式の検索結果の比較を行なう．但し，Cleverproject は検索結果を公表していないので直接結果の比較を行なうことはできない．よって，アルゴリズム上の比較を行なう．

Cleverproject と本方式のアルゴリズムの最大の違いは最終的な結果を求める時に利用するグループが異なる点である．

Cleverproject は既存の検索システム（Altavista 等）の結果とそれのリンク／被リンクサイトを合わせた集合であるルートセットを検索対象グループとしている．ここで，Altavista 等は適合率の低いロボット型検索システムなので，検索キーワードに関連の低いサイトも結果に含まれることが予想される．そして，検索キーワードに関連のないサイトのリンク／被リンクサイトが検索キーワードに関連するものである可能性は低い．また，検索キーワードに関連のあるサイトのリンク／被リンクサイトが検索キーワードに関連する保証もない．

それに対して，本方式ではリンク集の集合（コミュニティ）のリンク先サイトを検索対象としている．そして，これらリンク集のリンク先サイトはそのトピッ

クに関する専門家が精選したものなので，より質の高いサイトであることが予想される．また，4.3 節で述べたように本方式のコミュニティが検索キーワードに関連するサイトへのリンクを複数持つ確率は 90 %であり，リンク集の質は保証されている．

これらを考慮すると，Web コミュニティ方式の方が質の高い検索対象グループを用いているので，より質の高い結果を返すことができると考えられる．

7. おわりに

本稿では，Web コミュニティ方式の情報検索システムについて述べた．本方式は URL とアンカー情報を利用してリンク集を得ることで，精選されたリンクを利用し高い適合率を発揮することができる．また，Google，Cleverproject などの既存のリンク分析型の情報検索システムとの比較考察を行ない，その問題点を改善できることを示した．今後は，自前のインデックスを持つことで検索処理の改善を行なう必要がある．

参考文献

1) NetcraftWebServerSurvey: http://www.netcraft.com/Survey/
2) Steve Lawrence and C. Lee Giles: How big is the Web? How much of the Web do the search engines index? How up to date are the search engines?, availavle at http://www.neci.nj.nec.com/homepages/lawrence/Websize.html
3) Google search: http://www.google.com
4) Sergey Brin and Lawrence Page: The Anatomy of a Large-Scale Hypertextual Web Search Engine. WWW7, Australia, Computer Networks 30(1-7), pp 107-117, Apr. (1998).
5) Junghoo Cho, Hector Garcia-Molina, and Lawrence Page: Efficient Crawling Through URL Ordering. In Proceedings of the Intl. WWW Conf., (1998). available at http://www-db.stanford.edu/ cho/crawler-paper/
6) IBM Almaden Research Center, available at http://www.almaden.ibm.com/cs/k53/clever.html
7) 久我 昌崇, Web コミュニティのリンク構造分析に基づく情報検索アルゴリズム, 情報処理学会第 60 回大会講演論文集（3）1U-6, pp 147-148, (Oct. 2000).
8) Yahoo Japan: http://www.yahoo.co.jp
9) goo Search: http://www.goo.ne.jp
10) Sergey Brin, Rajeev Motwani, Lawrence Page, Terry Winograd: What can you do with a Web in your Pocket? IEEE Computer Society, Bulletin of the Technical Committee on Data Engineering, Vol. 21 No. 2, pp.37-47, Jun. (1998).
11) Soumen Chakrabarti, Byron Dom, S. Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, Andrew Tomkins, Jon M. Kleinberg, David Gibson: Hypersearching the web. Scientific American, Jun. (1999). available at http://www.sciam.com/1999/0699issue/0699raghavan.html
12) S. Chakrabarti, B. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. Experiments in topic distillation. SIGIR workshop on Hypertext IR (1998).
13) Altavista:http://www.altavista.com/
14) Ravi Kumar, Prabhaker Raghavan, Sridhar Rajagopalan, Andrew Tomkins: Ttawling the Web for emerging cyber-communities. In Proc. of 8th WWW Conf. (1999). available at http://www8.org/w8-papers/4a-search-mining/trawling/trawling.html
15) Ravi Kumar, Prabhakar Raghavan, Sridhar Rajagopalan, D.Sivakumar, Andrew Tomkins, Eli Upfal: The Web as a graph. Proc. of the 19th ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems (2000).
16) Sourav S. Bhowmick, Sanjay K. Madria, Wee-Keong Ng, Ee-Peng Lim: Web Bags-Are They Useful in A Web Warehouse?, FODO'98, Japan, Nov. (1998), available at http://www.cais.ntu.edu.sg:8000/tr/tr9813.ps
17) Infonavigator: http://infonavi.infoWeb.ne.jp/option.html
18) Cybeillance: New Cyveillance Study Fueled by NetSapien Technology Projects Size of the Internet Will Double in Less Than a Year, Jun. (2000). available at http://www.cyveillance.com/newsroom/pressr/000710.asp
19) Reka Albert, Hawoong Jeong, Albert-Laszlo Barabasi: Attack and error tolerance in complex networks, Nature in press. (2000). available at http://www.nd.edu/ networks/Papers/attack.pdf
20) Albert-Laszlo Barabasi, Reka Albert: Emergence of scaling in random networks, Science 286 pp.509-512 (1999). available at http://www.nd.edu/networks/slide/table.html
21) Andrei Broder, Steve Glassman, Mark Manasse, and Geoffrey Zweig: Syntactic clustering of the Web. In Proc. of 6th WWW Conf, pp.391-404, Apr. (1997). available at http://gatekeeper.dec.com/pub/DEC/SRC/technical-notes/SRC-1997-015-html