

半構造データを用いた WWW リンク構造解析アルゴリズムの新提案

A New Algorithm for Analyzing the Link Structure of WWW Using
Semi-structured Data

<http://www.tani.cs.chs.nihon-u.ac.jp/g-2005/yumiko/>

谷 研究室 西村 有美子

Yumiko NISHIMURA

概要

Web ページや XML 文書に代表される半構造データから、効率よくより良い情報を抽出する技法が期待されている。本研究では、WWW のリンク構造をグラフとみなし、最適な情報を抽出する方法として知られている HITS アルゴリズムに、テキスト解析を加えた改善手法を提案し、その有効性を検証するための比較実験を行う。

1 はじめに

今日のインターネットの普及に伴い、World Wide Web は、情報を入手する目的で多くの人々に利用されている。この Web 上の情報はコンピュータネットワークの発達と共に増え続け、膨大な量の情報の中から有用な情報を見つけることは利用者にとって困難になるばかりである。そこで、こうした状況の中で、情報を扱いやすくする為の方法が多く試みられている。その 1 つに、ある共通するトピックを持った Web ページ集合“Web コミュニティ”のリンク構造に注目した HITS アルゴリズム [1] がある。この HITS は非常に有名であるが、ページ間のリンク関係だけを利用して Web ページの重み付けを行っており、各 Web ページのコンテンツに立ち入らないことが前提となっている。つまり、探したいトピックに関係のないページも関係のあるページと同等に扱ってしまう。そこで本研究では、Web ページの内容を考慮した重み付けを行うため、HITS のリンク構造解析に、Web ページが“半構造データ”であることを利用したテキスト解析を加えた改善手法を提案した。また、その有効性を評価するための比較実験を行った。

以下、第 2 章で HITS アルゴリズムの手法を紹介し、第 3 章では HITS アルゴリズムの問題点、及び改善手法を記し、第 4 章で比較実験、評価結果を述べ、まとめる。

2 HITS(Hyperlink-Induced Topic Search)

この章では、Web 上のリンク構造を解析し、Authority と Hub と呼ばれる 2 つのページ集合から信頼性の高い情報を抽出した、Jon M.Kleinberg の HITS (Hyperlink-Induced Topic Search) アルゴリズムを紹介する。

2.1 HITS アルゴリズムの概要

HITS アルゴリズムの特徴は、各 Web ページのコンテンツに立ち入らずページ間のリンク関係を解析することにより、検索トピックに対して適切な情報 (“Authority” と “Hub” の 2 タイプからなる Web コミュニティ) を抽出するところにある。HITS アルゴリズムにおける “Authority” とは、特定のトピックにおける的確な情報を持つと承認されるページ群を意味する。また “Hub” とは、多くの Authority ヘリンクを張っているリンク集的なページ群である。

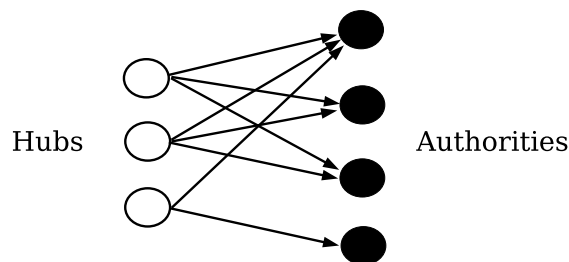


図 1: Hub と Authority の関係

2.2 HITS アルゴリズムの手順

STEP1

探したいトピックをサーチエンジンに投入し、一定数 r 件の Web ページを収集し、root set とする。

STEP2

root set に含まれるページからリンクされているすべてのページ、及び root set に含まれるページにリンクしているページ 1 件につき最大 d 件を収集し、root set に追加して大きさ n の base set を作成する。

STEP3

base set の全てのページについて, Authority や Hub に重みを付けいていく.

なお, Kleinberg は 各パラメータを $r = 200, d = 50$ に設定し, その結果 n は約 1000~5000 になると報告している.

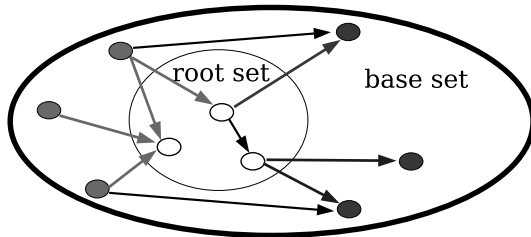


図 2: root set と base set の関係

2.2.1 Authority と Hub の重みのつけ方

Authority と Hub は, 各ページ p に与えられた Authority weight x_p と Hub weight y_p を計算する次の 2 式を反復することにより抽出される.

$$x_p = \sum_{q \text{ s.t. } q \rightarrow p} y_q$$

$$y_p = \sum_{p \text{ s.t. } p \rightarrow q} x_q$$

つまり, 両者の関係は「よい Authority は複数の良質の Hub によってリンクされ, また, 良質の Hub は複数のよい Authority にリンクを張っている」と再帰的に定義している. 結局これらは, ページ間の隣接行列に関連する固有値問題である.

隣接行列

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix}$$

$$a_{ij} = \begin{cases} 1 & \text{if page } i \text{ pointing to } j \\ 0 & \text{otherwise} \end{cases}$$

反復計算

$Z = (1, 1, \dots, 1) \in R^n$ とし, $X_0 = Z, Y_0 = Z$ とする. そして更新ルールを

$$\begin{cases} X_i = A^T Y_{i-1} \\ Y_i = A X_i \end{cases}$$

として, 上記の 2 式の操作と正規化を繰り返し行い, X と Y の極限 X^* と Y^* を得る. また, 上記の 2 式より, 反復回数を k とし, $i = (1, 2, \dots, k)$ に対して

$$\begin{cases} X_k = (A^T A)^{k-1} A^T Z \\ Y_k = (A A^T)^k Z \end{cases}$$

の 2 式が導かれる. $A^T A, A A^T$ は $n \times n$ の対称行列であるため, X^* は $A^T A$ の最大固有値に対応する固有ベクトル, Y^* は $A A^T$ の最大固有値に対応する固有ベクトルである.

3 HITS アルゴリズムの改善

この章では, HITS アルゴリズムの問題点を挙げ, 改善手法を提案する.

3.1 HITS アルゴリズムの問題点

HITS アルゴリズムは, 各ページからのリンクは人為的に関係付けられたものであり重要なページへのリンクが多い, という前提のもとで行われている. しかし, 現在の Web コミュニティは必ずしもそうとは限らない. つまり, 内容の異なったページにリンクしていることも多々あるということである. そのため, base set に検索トピックとは無関係な, あまり適切ではないページが含まれ, そうしたページが密なリンク構造である場合, 探したいトピックとは関係のないページの weight が高くなってしまふ. これが topic drift 問題である.

3.2 HITS アルゴリズムの改善点の概要

HITS アルゴリズムは, 各 Web ページのコンテンツに立ち入らないことが前提となっており, 探したいトピックに関係のないページも関係のあるページと同等に扱ってしまう. そこで本研究では, Web ページが“半構造データ”であることを利用した改善手法を提案する. 半構造データとは, 無構造データと構造データの間のデータ構造のことである. 無構造データとはスキーマ (データベースで論理構造や物理構造を定めた仕様) が全く存在しない (例えば, 普通の文章などの) データのことであり, 構造データとは, スキーマが厳密に決まっている RDB (Relational Data Base) のようなデータのことである. つまり, 半構造データとは, タグによって部分的に構造を記述できるデータのことであり, 代表的な例として XML や HTML が挙げられる.

3.2.1 改善手法

以下に半構造データを利用した改善手法を 2 つあげる.

(1) tag weight の導入

ページの内容に立ち入り, 探したいトピックが特定の HTML タグで強調されているならば, その頻度を隣接行列に反映させる. 本研究では, 特定の HTML タグとして“ $\langle B \rangle$ ”“ $\langle I \rangle$ ”“ $\langle META \rangle$ ”“ $\langle STRONG \rangle$ ”“ $\langle TITLE \rangle$ ”“ $\langle Hx \rangle$ ($x = 1, 2, 3, 4, 5$)”を使用した.

(2) anchor weight の導入

ページの内容に立ち入り, ページ内のアンカーテキスト, アンカータグ, 及びアンカータグ前後のテキストに探したいトピックがあれば, その頻度を隣接行列に反映させる. ここでアンカータグは“ $\langle A \ HREF = \text{"http://www..."} \rangle \sim \langle /A \rangle$ ”である.

(1) tag weight の導入は, ページ内で探したいトピックが HTML タグによって強調されているということは, そのページはトピックと関係していると思われ, ページの weight を重くしても良いと考えられるからである. しかし, Web ページは個人的なものも多く, 開始タグのみで終了タグのない場合や, タグ本来の意味とは異なった意味で使われている場合など, きちんとしたタグの使い方がなされていない場合も多々ある.

そこで (2) anchor weight の導入では, ページ内のアンカーテキスト, アンカータグ, 及びアンカータグ前後のテキストにはリンク先のページに関する説明が含まれている可能性が高いと考えられるので, この手法を提案する.

3.3 トピック値

前項の (1), (2) に当てはまる場合, ページに一定の値 (トピック値) を与える.

3.3.1 トピック値の決定

トピック w に対し, ページ p における w の頻度を w_p とし, トピックベクトル W を作成する.

$$W = \begin{bmatrix} w_1 \\ \vdots \\ w_n \end{bmatrix}$$

w_p = ページ p におけるトピック値

3.3.2 ページ間の隣接行列の改善

2.2.1 で示したページ間の隣接行列を次のように改善する.

$$A = \begin{bmatrix} a_{1,1} & \cdots & a_{1,n} \\ \vdots & & \vdots \\ a_{n,1} & \cdots & a_{n,n} \end{bmatrix}$$

$$a_{ij} = \begin{cases} 1 + w_j & \text{if page } i \text{ pointing to } j \\ 0 & \text{otherwise} \end{cases}$$

以降は 2.2.1 と同様の手順で進める.

4 比較実験

4.1 実験方法

今回の実験では, いくつかのトピックに対し, 以下の 4 つの方法を用いて比較実験を行った.

- (a) Google サーチエンジン [2]
- (b) HITS アルゴリズム
- (c) HITS に tag weight を導入したもの
- (d) HITS に anchor weight を導入したもの

上記の (b)(c)(d) の 3 つのアルゴリズムにおいては, 各パラメータを $r = 100, d = 50$ に設定し, root set の収集は Google Web API [3] を用いて行った. Google Web API とは, 開発者が自分専用のインターフェイスをつくり, Google の豊富で膨大な量のデータベースを利用して検索結果を好きなように利用することが出来る API のことである. また, API とは Application Programming Interface の略で, 特定の資源やアプリケーション, この場合は Google のデータベースにプログラムを使ってアクセスする際の入口のようなものである. 本研究で使用する root set 100 件は Google サーチエンジンでトピックを検索した時の上位 100 件を使用している.

4.2 評価方法

今回, 評価の方法として, 定性的調査 (グループインタビューのように少ない標本を対象にした調査で, 仮説の設定, 関連性・因果関係の定性的な解明・検証, アイデア・コンセプトの開発等に用いられる調査) をアンケート形式で行った. アンケートの調査方法は, Jon M.Kleinberg の HITS アルゴリズムの調査を行った研究グループを参考にした. まず調査対象者の選び方として“web ブラウザに精通している人”という事で, 本来

ならばランダムに対象者を選ばなくてはならないが、時間の関係上、谷研究室 20 名を対象にアンケート調査を行った。アンケートの方法は、21 個のトピックにおいて (a)(b)(c)(d) の方法でそれぞれの結果の上位 3 つの web ページを閲覧し、その内容を、“bad ” fair ” good ”の 3 段階で評価してもらった。また、(a)(b)(c)(d) のどの方法がどの web ページを返したかは伏せて行った。

4.3 実験結果

4 つの方法で比較した結果、各々、次のような支持率になった。

- (a) Google サーチエンジン：29.91%
- (b) HITS アルゴリズム：12.64%
- (c) HITS に tag weight を導入したもの：30.83%
- (d) HITS に anchor weight を導入したもの：26.62%

また、具体的な結果については別途表を参照されたい。

4.4 考察

今回、Web ページが半構造データであることを利用した HITS アルゴリズムの改善手法を 2 つ提案し、実験に用いたトピックにおいては、ほぼ全てに関し (HITS アルゴリズムに対しては) 改善を見ることが出来た。このことから、

- ・各 Web ページのコンテンツに立ち入って、ページを評価することは有効である

といった知見が得られた。

また、改善手法の中でも tag weight の導入の方が良い結果が得られたことから、

- ・HTML タグのによって強調されている言葉は、そのページのキーワードとして捉えることが出来る

といったことも得られた。

そして、わずかではあるが、Google サーチエンジンより良い結果を得ることが出来た要因として、Google は、どんなトピックを検索しても、上位の結果が企業のページになることが多いことが挙げられる。しかし、圧倒的に Google が支持されるトピックの種類があった。それは、企業名や大学名などの固有名詞である。固有名詞をトピックとしたとき、Google では、その正式なホームページが上位に現れる。

以上の考察を踏まえて、提案手法に更なる改善を施す

ことによって精度向上の可能性がある、ということを提案としてまとめとする。

4.5 今後の課題

考察において提案手法の更なる改善を提案したが、ここで、具体的な改善方法を述べる。

まず、ページの URL にトピックが含まれ、さらにドメイン名が“ co.jp ”や“ ac.jp ”等であった場合、特別にページの weight を重くしてみると Google サーチエンジンに比べ、明らかに劣っていた点で改善が期待できるかもしれない。

また、今回 tag weight の導入で使用した HTML タグによる、ページへの weight は全て同じにしたが、“ < TITLE > ”や“ < H1 > ”のタグによってトピックが強調されていた場合、与える weight を重くするなど、タグの種類によって weight の重みを変えることで、よりトピックに関係したページが上位に現れるかもしれない。

最後に、考察では特に挙げなかったが、入力するトピックを 1 つに限らず、複数のトピックで検索してみると、また違った結果を得ることが出来るのではないかと思う。

参考文献

- [1] John M.Kleinberg, “Authoritative Sources in a Hyperlinked Environment,” Research Report RJ 10076, IBM. 1997.
- [2] <http://www.google.co.jp/>
- [3] Google Web API, <http://www.google.com/apis/>.
- [4] Tara Calishain, Rael Dornfest, Google Hacks, O'Reilly Japan.
- [5] Ian F.Darwin, Java Cookbook, O'Reilly Japan.
- [6] Elliotte Rusty Harold, Java Network Programming 2nd Edition, O'Reilly Japan.
- [7] Jama, <http://math.nist.gov/javanumerics/jama/>.
- [8] 武田 善行, 梅村 恭司, 藤井 敦, Web マイニング, 共立出版, 2004.
- [9] 北 研二, 津田 和彦, 獅々堀 正幹, 情報検索アルゴリズム, 共立出版, 2002.

実験結果

(1) tag weight の導入が一番良かったもの

topic : orangepekoe

・ Google サーチエンジン

順位	URL
1	http://homepage2.nifty.com/orangepekoe/
2	http://homepage2.nifty.com/orangepekoe/editor/editor.htm
3	http://orange-pekoe.64b.org/

・ HITS アルゴリズム

1	http://boards.fool.co.uk/EditProfile.asp
2	http://boards.fool.co.uk/EditFavoriteFools.asp?love=15388&page=p
3	http://boards.fool.co.uk/EditFavoriteFools.asp?ign=15388&page=p

・ tag weight を導入したもの

1	http://www.amazon.co.jp/exec/obidos/ASIN/B00061QXU4
2	http://groups.yahoo.co.jp/group/fukuoka-orangepekoe/
3	http://bonten.homeip.net/ol/

・ anchor weight を導入したもの

1	http://hpcgi2.nifty.com/orangepekoe/sitemap/sitemap.cgi
2	http://groups.yahoo.co.jp/group/fukuoka-orangepekoe/
3	http://groups.yahoo.co.jp/group/orangepekoe/

(2) anchor weight の導入が一番良かったもの

topic : 世界遺産

・ Google サーチエンジン

順位	URL
1	http://www.tbs.co.jp/heritage/
2	http://www.tbs.co.jp/heritage/index-j.html
3	http://homepage1.nifty.com/uraisan/

・ HITS アルゴリズム

1	http://homepage1.nifty.com/uraisan/yakata/a01.html
2	http://homepage1.nifty.com/uraisan/yakata/a02.html
3	http://homepage1.nifty.com/uraisan/yakata/a03.html

・ tag weight を導入したもの

1	http://homepage1.nifty.com/uraisan/rireki.html
2	http://www.dango.ne.jp/sri/whinformation.html
3	http://www.dango.ne.jp/sri/shucchoukouza.html

・ anchor weight を導入したもの

1	http://www.dango.ne.jp/sri/whinformation.html
2	http://homepage1.nifty.com/uraisan/rireki.html
3	http://www.morisawa.org/db_kyoto.html