

非巡回有向グラフによる フォークソノミータグの局所拡張可能な配置方法

江田 毅晴[†] 吉川 正俊^{††} 山室 雅司[†]

[†] NTT サイバースペース研究所

^{††} 京都大学大学院情報学研究科

E-mail: [†]{eda.takeharu,yamamuro.masashi}@lab.ntt.co.jp, ^{††}yoshikawa@i.kyoto-u.ac.jp

あらまし 本研究では、フォークソノミーデータの分析に基づく、タグの階層的配置方法について提案する。現在のソーシャルブックマークサービスでは、大量のブックマークエントリの中から有用なリソースを探すには、キーワードやタグを指定するか、興味の近いブックマークを努力して探し出す必要がある。本研究では、タグの共起関係に基づく意味的な繋がりを PLSI を用いて解析し、タグを非巡回有向グラフに配置する方法を提案する。既存手法である木構造による配置方法に比べて、子ノードの数を担保でき、また目的のノードに到達するためのパスを複数用意することができる。これにより、タグを利用したユーザの誘導を容易にすることが可能となる。

キーワード フォークソノミー, ソーシャルブックマーク, 分類, タギング

Locally Expandable Allocation of Folksonomy Tags in a Directed Acyclic Graph

Takeharu EDA[†], Masatoshi YOSHIKAWA^{††}, and Masashi YAMAMURO[†]

[†] NTT Cyber Space Laboratories

^{††} Department of Social Informatics, Kyoto University

E-mail: [†]{eda.takeharu,yamamuro.masashi}@lab.ntt.co.jp, ^{††}yoshikawa@i.kyoto-u.ac.jp

Abstract We propose a new classification system based on an analysis of folksonomy data. To find valuable resources from current social bookmark services, users need to specify search terms or tags, or to discover people with similar interests. Our system uses semantic relationships extracted from the co-occurrences of folksonomy data using PLSI and allocates folksonomy tags in a directed acyclic graph. Compared to the hierarchical allocation method of a tree, our method guarantees the number of children nodes and increases the number of available paths to an objective node, enabling users to navigate the resources using tags.

Key words Folksonomy, SBM, Classification, tagging

1. はじめに

ブログやウィキといったユーザ参加型のウェブサービスが普及している。これらのサービスの大きな特徴として、書き込みの頻度が非常に高く、ページ間のリンクを半自動的に生成するオートリンクが大量に存在しており、結果としてウェブリンクの価値を低下させているという状況がある。また、ブログに対するスパムであるスプログも大量に発生しており、機械的にリンクを解析して Web コンテンツをランキングすることがますます困難になっている。

昨今、注目を集めているソーシャルブックマークサービス (SBM) (del.icio.us [2], はてなブックマーク [11], など) はこう

した問題を解決するための 1 つの手段と捉えることができる。SBM は、ボトムアップな分類システムであるフォークソノミー (folksonomy) を採用しており、ユーザにブックマーク機能を提供し、サーバサイドにブックマークエントリを格納する。ユーザはブックマーク時に複数個のタグを付与することができ、コメントを付与することもできる。集められたブックマークエントリを利用することにより、新着リソースのランキングを行うことができる。また、ユーザはコメントやブックマークのつながりを利用して緩いコミュニティを形成している。ユーザが低コストでサービスを利用し始めることができる点も、SBM がリソース分類システムとして人気を集めている理由の 1 つと考えられる。フォークソノミーにおいては、分類の専門家を雇う人

件費や、時間をかけてリンク解析をする必要もないため、ブログエントリ等の鮮度の高い情報の分類には非常に適している。こうした意味で、ソーシャルブックマークサービスは、次世代のウェブリソースのランキングおよび分類方法の有力候補の一つと考えられる。

しかしながら、フォークソノミーではユーザが任意のタグを付与することができるため、類義タグ、多義タグの存在や、タグの種類が爆発するといった問題がある。たとえば、“web”、“internet”、“network”といったタグの間に関連性があることをユーザならば直感的に認識できるが、計算機が外部の知識無しに自動的に認識することは困難である。通常のSBMで用いられているタグクラウドと呼ばれる表示形式では、タグが五十音順・頻度順で並べられているが、タグの意味的な繋がりが順序に反映されないためユーザが望むタグを探しだすのは容易ではない。

そこで本研究では、タグの関連性を抜き出し、関連性を利用してタグを階層的に配置する手法を提案する。提案手法では、タグを共起関係に基づき特徴ベクトル化し、特徴ベクトル間の距離および曖昧性を推測することによって、非巡回有向グラフ(DAG(Directed Acyclic Graph))に配置する。これにより、ユーザによるタグの繋がりを利用した情報取得が容易になることが期待できる。本研究の貢献は以下の通りである。

- DAGに基づいたフォークソノミータグの階層的配置方法を提案する。提案手法では、文献[10]にて提案されているPLSI(Probabilistic Latent Semantic Indexing)によって計算した確率ベクトルを基にタグの距離を測定し、確率ベクトルの曖昧性の尺度であるエントロピー値を用いてタグの上位下位関係を構築する。

- DAGを利用した階層による利点・欠点を整理し、既存手法であるタグを木構造に配置する手法に対して優れている点を、実験を通して定性的に分析する。

本論文は以下のように構成される。

まず、2.章にて関連研究について述べ既存手法の問題点を明らかにする。3.章にて提案手法について説明し、4.章で定性的な評価を行う。最後に、5.章にてまとめと今後の課題について述べる。

2. 関連研究

多数のフォークソノミーに関する研究が発表されている[9]。BrooksとMontanez[1]は、本文中からのキーワード抽出による自動タグ付けとタグの階層化について議論している。階層化されたタグはオントロジーとしてとらえることができるため、様々なアプリケーションが提案できるとしているが、同時に分類体系としては木構造には柔軟性がないことも指摘している。

GolderとHuberman[3]は、del.icio.usにおけるタグ、ユーザ、リソースの分布を分析しており、各人によってブックマークの傾向にかなりの偏りがあることを示している。Sen等[8]の報告は、MovieLensシステムにて協調的分類システムを実装したケーススタディになっており、タグの整理は人手で行われている。丹波等[12]は、Webページ推薦を目的としてフォークソノ

ミーを利用している。類義語、多義語の問題はクラスタリングを行うことによって解決できるとしているが、タグの分類システムとしての評価は行っていない。

同義タグ・多義タグの問題に関して、Xian等[10]は、PLSIを用いて、フォークソノミーの三つ組(ユーザ、タグ、リソース)の索引付けを行い、確率モデルに基づく新しい検索モデルを提案している。Xian等の手法の最大の利点は、タグ、ユーザ、リソースの全てが同等に索引付けされることにある。

タグを機械的に整理するという観点で問題を捉えるのであれば、タグをクラスタリングすることが考えられる。しかしながら、クラスタリングでは階層構造の中間タグを決定することが困難である。

Heymann等[5]は、フォークソノミーベクターモデルを提案し、タグを木構造に配置する方法として、拡張グリーディ法(Extensible Greedy Algorithm)を提案している。フォークソノミーベクターモデルでは、それぞれのタグは、URLに対してそのタグでブックマークを行ったユーザの人数を値とするベクトルで特徴づけられる。特徴ベクトルを抽出後、タグ間の距離をコサイン類似度で測り、タグ全体をネットワークと見なして、それぞれのタグの中心性(Centrality)を計算する。中心性とは、ノードのネットワークの中での中心への近さを表す尺度である。Heymannらの手法では、この中心性の順にタグをソートし、トップダウンに木を構築していく。しかしながら、木構造でタグを配置するとノードを必ずある1つの親の子供として配置しなければならないという問題があり、そのようにして決定された木構造には、不自然な箇所が発生してくる。通常のカテゴリ体系であるオープンディレクトリ^(注1)やGoogleディレクトリ^(注2)を観察すればノードが複数の親を持つのはむしろ自然である。また、フォークソノミーベクトル空間が広大でまばらであるということも問題である。すなわち、多くのベクトル間の類似度が0になり、葉ノードに近づくほど精度が下がってくる。図1に、Heymann等の手法を用いて作成した木構造のうち、“mac”を根とする部分木を示す^(注3)。葉ノードに近づくほど、“mac”というタグの持つ意味とかけ離れた意味を持つタグが登場してくるのが分かる。

3. 提案手法

本論文の提案手法では、PLSIを用いてタグの特徴ベクトルを取得するが、そのようにして取得したベクトルをPLSIベクトルと呼ぶことにする。今回は、Xian等によって提案された、ユーザ、タグ、リソース(URL)の三つ組をモデル化したPLSIを利用した[10]。PLSIを用いると、タグだけでなくユーザ、リソースも同時にPLSIベクトルとして抽出できるが、本研究ではタグのPLSIベクトルのみを利用している。以降、記述上はタグをそのタグのPLSIベクトルと同一視することにする。PLSIベクトルを解析して、DAGにタグを配置する。処理全体の流れを、図2に示す。最初に、ブックマークエントリの三つ組に対

(注1): <http://www.dmoz.org>

(注2): <http://directory.google.com/>

(注3): 4.章の実験で用いたデータセットを用いて可視化したもの

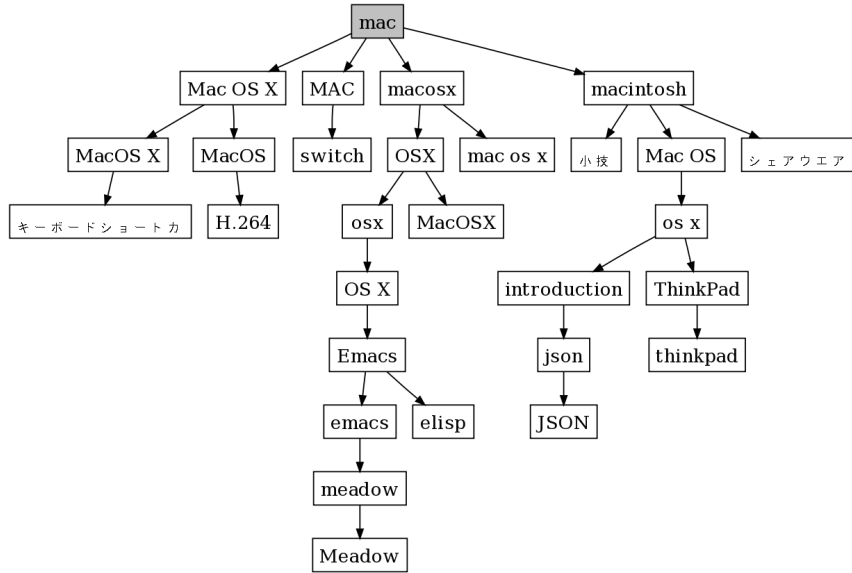


図 1 拡張グリーディ法による木. (EXT 手法)

Fig. 1 Heymann's Tree (EXT Approach).

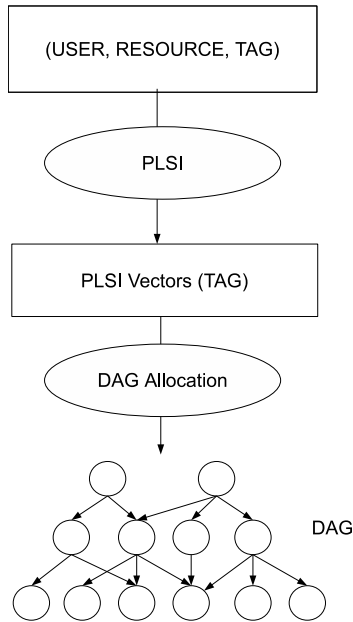


図 2 全体の処理の流れ.

Fig. 2 Overall Processing Flow.

して、PLSI による索引付けを行い、PLSI ベクトルを取得する。これらの PLSI ベクトル間の距離を JS ダイバージェンスで計測し、タグの上位下位判定にエントロピー値を用いることにより、タグを DAG 状に配置する。PLSI はオフラインで計算する必要があるが、DAG 配置は局所的にオンラインで行うことができる。

3.1 PLSI を用いたフォークソノミーエントリの索引付け

PLSI を用いると、複数のアイテム集合の間での共起関係データ (Co-occurrence Data) から、与えられた次元数の確率値ベクトル空間へそれぞれの集合のアイテムを射影することができる。PLSI は二つ組である (文書 ID, ワード) の共起データに対して提案されたが [6], その後, Xian 等によってフォークソノミーの

3 組共起データへと拡張されたものが提案されている [7]. PLSI ベクトル間の距離を測定することにより、ベクトル間の近さをあらかじめ索引付けすることができる。PLSI のモデルでは、アイテム間の潜在的意味空間を想定しているため、PLSI ベクトルの近さは潜在意味空間における近さになる。

図 3 に三つ組 PLSI におけるモデルを示す。ユーザ、リソース、タグそれぞれのアイテム集合を U, R, T とする。今、射影したい潜在的意味空間の次元数を D , ある次元 $\alpha \in [1, 2, \dots, D]$ からユーザ $u_i \in U$, リソース $r_j \in R$, タグ $t_k \in T$ の生起確率を、それぞれ $p(u_i|\alpha), p(r_j|\alpha), p(t_k|\alpha)$ とする。また次元 α の生起確率を $p(\alpha)$ とすると、 u_i, r_j, t_k の間には、潜在的意味ベクトル空間を介して以下の関係式が成り立つ。但し、 $p(u_i, r_j, t_k)$ は、三つ組の共起確率である。

$$p(u_i, r_j, t_k) = \sum_{\alpha=1}^D p(\alpha)p(u_i|\alpha)p(r_j|\alpha)p(t_k|\alpha)$$

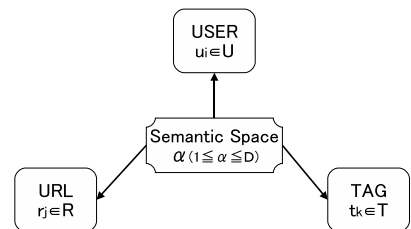


図 3 ユーザ、URL、タグの間のモデル.

Fig. 3 Probabilistic Model Among User, URL, and Tag.

左辺はデータセットから観測できる値であり、右辺の確率値は適当な初期値を与えることによって、EM アルゴリズムによって計算可能である。EM アルゴリズムは E (Expectation) ステップと M (Maximization) ステップからなり、これらを交互に逐次繰り返すことによって、最尤推定を行う。

次に推定された確率値に対しベイズ則を用いて、各潜在意味

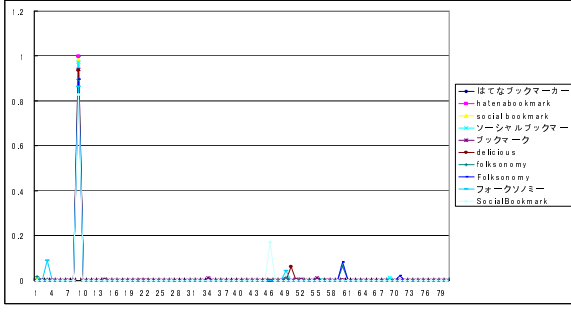


図 4 PLSI ベクトルの例.
Fig. 4 Example of PLSI Vectors.

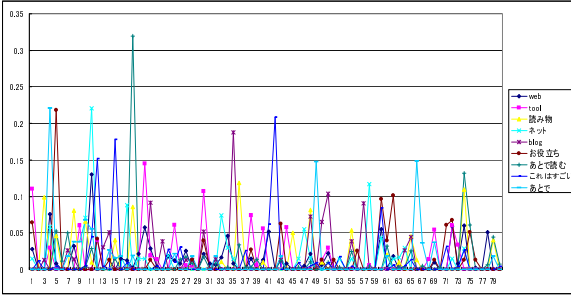


図 5 エントロピー値の高い PLSI ベクトルの例.
Fig. 5 High Entropy PLSI Vectors.

次元からユーザ, リソース, タグへの条件付き確率を計算する. α 次元からの条件付確率を, それぞれのアイテムの α 次元の値とすることによって, すべてのアイテムを D 次元ベクトルとして表現できる.

3.1.1 PLSI ベクトルの観察

図 4 に実際に計算した PLSI ベクトルの例を示す. 横軸が次元であり, 縦軸がその次元の確率値である. 図 4 にある PLSI ベクトルは, 主に「ソーシャルブックマーク」に関するものを表示しているが, 全てのベクトルが第 9 次元に強く特徴を持っていることが分かる.

PLSI ベクトルのエントロピー

タグ t_k のエントロピー値とは, 以下の式で計算される値である(注4).

$$H(t_k) = - \sum_{\alpha=1}^D p(\alpha|t_k) \cdot \log(p(\alpha|t_k))$$

エントロピーの値は, PLSI ベクトルが 1 つのピーク (最大値 = 1) を持つ場合には 0 をとり, 一様分布に近いほど高くなり, PLSI ベクトルの曖昧性を表していると考えられることができる. 図 5 には, 全 PLSI ベクトルの中から, エントロピー値が高い PLSI ベクトル 9 個を示している. 図 6 は, 逆にエントロピー値が 0 である PLSI ベクトルを表示している.

PLSI ベクトル間の距離

本研究では確率値ベクトル間の距離尺度として有効であるとされる JS ダイバージェンスを用いる. JS ダイバージェンスは, 下記の式で計算される(注5).

(注4): ユーザ, リソースにも同様に定義される.
(注5): $D(q||r)$ は KL ダイバージェンス.

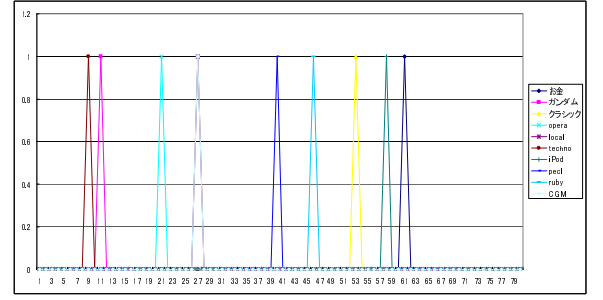


図 6 エントロピー値が 0 の PLSI ベクトルの例.
Fig. 6 Low Entropy PLSI Vectors.

$$D_{js}(t_k, t_l) = \frac{1}{2} [D(t_k || \text{avg}(t_k, t_l)) + D(t_l || \text{avg}(t_k, t_l))]$$

PLSI をタグの特徴付けに利用するメリットとしては, 通常の距離尺度では測れないが潜在的に存在するタグ間の関連を抽出できることが挙げられる. また, 全てのタグを低次元のベクトル空間に射影することができるため, ベクトルの扱いが容易になるという点もある.

3.2 DAG 配置アルゴリズム

3.2.1 DAG

DAG は, 閉路を含まない有向グラフである. 任意のノードから枝を辿ることによりノードを探索すると, 有限ノード数のグラフであれば, 必ず端点に辿り着く. 直感的には, DAG はノードが「一方へ」流れる」と理解することができる. 図 7 に, 提案手法にて構築した DAG の例を示す.

この「流れ」を, PLSI ベクトルのエントロピー値の大小によって決めることにより, PLSI ベクトルを特徴量として持つタグを DAG として配置することが可能になる. 表 1 に, 提案手法における DAG の構築アルゴリズムを簡潔に示す. 全て

表 1 DAG 構築アルゴリズム.

Table 1 DAG Allocation Algorithm.

```

input: PLSI ベクトルのリスト  $\{v_i\}_{i=1}^m$ 
output: PLSI ベクトルからなる DAG 構造
1 for( $i = 1; i < m + 1; i++ = 1$ )
2   set_children( $v_i, threshold$ )
3 endfor

```

関数: set_children

input: 親ノード: v , 閾値: $threshold$, 個数: k

```

1  $\{w_i\}_{i=1}^k : v$  近傍において距離が  $threshold$  以下かつ,  $k$  番目までのノードを取得
2 for( $i = 1; i < k + 1; i++ = 1$ )
3   if( $w_i.entropy < v.entropy$ ) then
4      $w_i$  を  $v$  の子ノードに設定
5   endif
6 endfor

```

のノードに対して, set_children 関数にて, 子供を設定する.

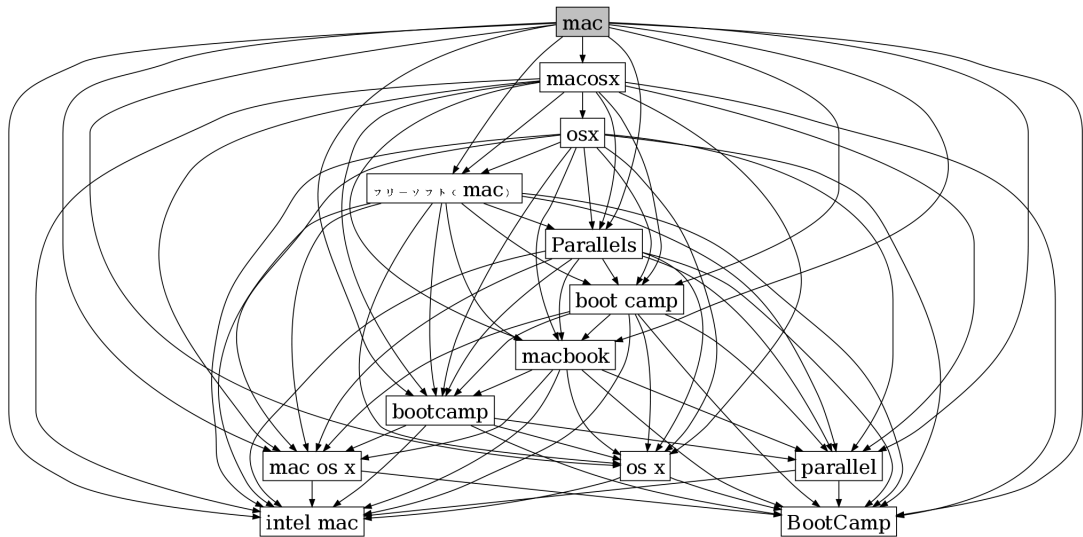


図 7 DAG (DAG 手法).

Fig. 7 DAG (DAG Approach).

set_children 内では、親ノードに対して閾値 *threshold* 以内の距離にありかつ上位 *k* 件に入るノードを探索し、親ノードのエントロピー値よりも低い近傍ノードを子ノードとして採用する。

子ノードとして親ノードよりもエントロピー値が低いもののみを選択しているため、あるノードを出発して順にノードを辿って行ったとき、ノードのエントロピー値は単調に減少し続ける。すなわち、同一のノードに返ってくることは不可能である。よって、表 1 で構築される有向グラフは必ず DAG になる。

表 1 のアルゴリズムでは、閾値 *threshold* と子ノード数 *k* の両方を指定して子ノードを選択している。両方を指定する理由は、PLSI 距離の値の分布によっては、子ノードが見つかり難いケースおよび子ノードが多すぎるケースもあるからである。*k* を適切に調整することにより、子ノードの数を担保することも可能となる。PLSI 距離の値の分布については 4. 章にて例を示す。

DAG の利点の 1 つとして、閉路を持たない点が挙げられる。すなわち、上位タグから下位タグに次々に辿っていったときに、同じところに戻ってくる事が無い。

提案する DAG 構築アルゴリズムの時間計算量は、*k* を十分小さいとみなすなら、 $O(n^2)$ (*n* はノード数) である。これは、関連研究 [5] の拡張グリーディ法と同等である。また、拡張グリーディ法が構造を一度全体として構築しなければ部分木を取得出来ないオフラインアルゴリズムであるのに対して、提案する DAG 構築アルゴリズムは任意のノードをキーとして動的に部分構造を展開することができる。

4. 評価

以下の項目を確認するために、評価実験を行った。

(1) PLSI ベクトルの近さと、フォークソノミーベクトルの近さに相関があるか。

(2) 2 手法でそれぞれ構築した分類構造において、あるタグの周辺構造がどのように異なるか。

項目 (2) は、フォークソノミーベクターモデルによる拡張グリーディ法 [5] と比較している。これ以降、PLSI ベクトルと DAG 配置アルゴリズムを利用した提案手法を DAG 手法、フォークソノミーベクターモデルによる拡張グリーディ法を EXT 手法と略記することにする。両手法の比較を表 2 に示す。データ構

表 2 DAG 手法と EXT 手法の比較。

Table 2 DAG and EXT Approaches.

	DAG 手法	EXT 手法
特徴ベクトル	PLSI ベクトル	フォークソノミーベクトル
距離尺度	JS ダイバージェンス	コサイン類似度
分類構造	DAG	木

造の可視化には、graphviz [4] を用いた。

データセットとしては、2006 年 10 月頃におけるはてなブックマークの人気エントリ集合を利用した。はてなブックマークのタグ一覧 (<http://b.hatena.ne.jp/t/>) にあるタグを取得し、そこからたどれる URL 集合から、三つ組みを手動で取得した。さらに、データベースに格納する際に、タグ、ユーザ、リソースとも、最低 5 回以上登場するエントリのみを利用した。その結果、各アイテムの情報は下記ようになった。

表 3 データセット。

Table 3 Data Set.

アイテム	個数
タグ	5496
ユーザ	11713
リソース	4019

PLSI および各種アルゴリズムのパラメータは、経験的に表 4 のように設定した。

表 4 実験時の各種パラメータ.

Table 4 Parameters.

DAG	PLSI の次元数	80
	PLSI の繰り返し数	80
	閾値: k	11
EXT	閾値: k	0.00001

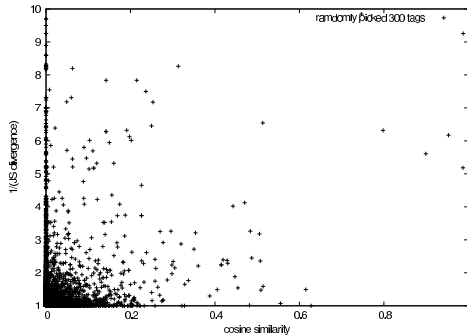


図 8 コサイン類似度と JS ダイバージェンス距離 (逆数) の分布.

Fig. 8 Distribution of Cosine and JS Divergence (inverse).

4.1 距離の違いについて

図 8 に、実験データ中から、300 種類のタグの対をランダムに選択し、そのタグ間のコサイン類似度を横軸、JS ダイバージェンスの逆数を縦軸にプロットしたものを示す。図から分かるように、フォークソノミーベクトルにおける近さと JS ダイバージェンスにおける近さには明らかな相関があるとは言えない。次に、表 5 にサンプリングしたタグ間の、コサイン類似度と JS ダイバージェンスの平均値を示す。類似度が低いということは

表 5 コサイン類似度と JS ダイバージェンスの平均値.

Table 5 Average Values of Cosine Similarity and JS Divergence.

	平均
コサイン類似度	0.0042
JS ダイバージェンス	0.9618

すなわち距離が遠いことを示すため、表 5 からは、平均してタグ同士は位置関係として端によっていることが分かる。また、細かい点を観察すると、コサイン類似度が 0 であるノードの JS ダイバージェンスが、広い範囲に分布していることに気づく。この点を確かめるため、コサイン類似度が最小になる場合と、JS ダイバージェンスが最大になる場合、すなわち両手法においてタグ間の距離が最も遠くなる場合の頻度を、ランダムに選んだ 87900 件の対から数えた。表 6 に示すように、フォークソノミー

表 6 距離が一番遠いケースの頻度 (87900 件中)

Table 6 Number of Maximally Distant Tag Pairs. (of 87900 pairs)

	頻度
コサイン類似度が 0	40315
JS ダイバージェンスが 1	6875

ベクトルを利用したコサイン類似度の場合、全体の約半数の距離が最も遠いと判定される。一方、PLSI を行った後に JS ダイバージェンスにて距離を測定すると、潜在的意味を通したつながりを利用することにより浮かび上がってくる関係がある。表 7 に、コサイン類似度が 0 で、JS ダイバージェンスが 1 に近いタグの対をランダムに選んだ例を示す^(注6)。表 7 は、コサイン類

表 7 コサイン類似度が 0 で JS ダイバージェンスが 1 に近い組合せの例.

Table 7 Example Tag Pairs Whose Cosine Equals 0 and JS divergence is near to 1.

rrdtool	アクセラレータ
楽天市場	Plone
CRC	partition
カースト	死刑
CSS3	メニュー
CSS3	xhtml
CSS ハック	メニュー
CSS ハック	CSS3
KDDI	Softbank
attention economy	nagios
フリーター	死刑
フリーター	カースト

似度では遠いと判定されたが、JS ダイバージェンスにより潜在的には近いと判定されたタグである。提案手法はこれらのタグ間の関係を抽出することができる。

4.2 局所的な構造の比較

いくつかのタグをサンプリングし、そのタグの子ノード一覧の比較を行った。サンプルした全てのタグについて、DAG, EXT 両手法について子ノードを取得し、手作業で精度を評価した。今回のデータセットでは同義語も多数含まれているため、同義語が登場した場合は正解であるとした^(注7)。タグの集合が膨大であるために再現率は評価していない。そのかわりに、子ノードとして提示されたノード集合の個数の平均値を示した。表 8 に両手法における結果を示す。DAG の平均精度は、EXT

表 8 子ノード数と精度の平均値.

Table 8 The Average Num. of Children and Precision.

	平均子ノード数	平均精度
EXT	3.00	0.75
DAG	8.00	0.64

に比べて 11% 低い。しかしながら、DAG が提示したノード数の平均は 8 であり、EXT に比べ 2.7 倍の個数を提示している。

表 9 に、いくつかのノードの子ノードの例を示す。拡張グリーディ法では、どこか一カ所にノードを配置するという制約があるため、特に葉ノードを適切に選択できないケースが多くなっているが、提案手法では閾値および k の指定により、子ノード

(注6): タグはランダムに選んだ。その他にも多数存在する。

(注7): 不正解とすると EXT 手法において子ノードを提示できる場合が減ってしまうため。

表 9 子ノードタグの例.

Table 9 Examples of Child Nodes.

親		子
database	DAG	DB , db , DataBase , sql , optimization , mySQL , 正規形 , dbms , SQL , Database , postgresql
	EXT	データベース, DB, DataBase, DBMS
mac	DAG	macosx , フリーソフト (mac) , boot camp , macbook , bootcamp , parallel , os x , mac os x BootCamp , intel mac , Parallels
	EXT	Mac OS X , MAC , macosx , macintosh
folksonomy	DAG	Folksonomy , sbs , del.icio.us , delicious , tags , SBM , sbm
	EXT	theory , Folksonomy

を柔軟に拡張して提示することが可能である。

典型的には、JS ダイバージェンス距離の特徴をいかした、拡張したつながりが登場するようになる。

4.3 考察

DAG 手法は、PLSI をあらかじめ行うことにより、フォークソノミーベクトルでは抽出することのできなかった潜在的に意味の近いタグを近傍ノードとして取得することができる。

DAG 手法は EXT 手法に比べて、11%の精度低下で 2.7 倍の子ノードを提示することができる。小さい精度低下で子ノードの数を増加させることができるため、提案手法はユーザにより多くの選択肢を提供するためのナビゲーションツールバーとして利用することが考えられる。

5. まとめと今後の課題

本論文では、DAG によるタグの階層的配置方法を提案した。提案手法では、PLSI を用いてフォークソノミーの三つ組データを解析し、タグを PLSI ベクトルとして特徴づける。PLSI ベクトルのエンтроピー値を用いてタグの曖昧性を推測し、近距離近傍取得と曖昧性の上下により、タグを DAG に配置する。実験結果により、既存研究である拡張グリーディ法に比べて、子ノードの精度の低下を 11%に抑えながら、2.7 倍の子ノードを提示できることを示した。

文献

- [1] Christopher H. Brooks and Nancy Montanez. Improved annotation of the blogosphere via autotagging and hierarchical clustering. In *Proc. WWW*, pp. 625–632, New York, NY, USA, 2006. ACM Press.
- [2] del.icio.us. <http://del.icio.us>.
- [3] Scott Golder and Bernardo A. Huberman. The structure of collaborative tagging systems. In *Journal of Information Science*, 2006.
- [4] Graphviz. <http://graphviz.org>.
- [5] Paul Heymann and Hector Garcia-Molina. Collaborative creation of communal hierarchical taxonomies in social tagging systems. Technical report. <http://heyman.stanford.edu/taghierarchy.html>.
- [6] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proc. UAI*, Stockholm, 1999.
- [7] Alexandrin Popescul, Lyle H. Ungar, David M. Pennock, and Steve Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *Proc. UAI*, 2001.
- [8] Shilad Sen, Shyong K. Lam, Al M. Rashid, Dan Cosley, Dan Frankowski, Jeremy Osterhouse, Maxwell F. Harper, and John Riedl. Tagging, communities, vocabulary, evolu-

tion. In *Proc. CSCW*, pp. 181–190, New York, NY, USA, 2006. ACM Press.

- [9] Jakob Voigt. Tagging, Folksonomy & Co - Renaissance of Manual Indexing? In *Preprint*, 2007. <http://arxiv.org/abs/cs/0701072v2>.
- [10] Xian Wu, Lei Zhang, and Yong Yu. Exploring social annotations for the semantic web. In *WWW*, pp. 417–426, 2006.
- [11] 株式会社はてな. はてなブックマーク <http://b.hatena.ne.jp>.
- [12] 丹羽智史, 土肥拓生, 本位田真一. Folksonomy マイニングに基づく web ページ推薦システム. 情報処理学会論文誌「マルチエージェントの理論と応用」特集号, Vol. 47, pp. 1382–1392, May 2006.