情報処理学会論文誌 Vol.53 No.12 2829–2840 (Dec. 2012)

RDFデータ検索のためのクエリグラフのクラスタリング手法

飯塚 京士[1,a)] 村山 隆彦[1,b)] 小林 透[2,c)] 赤埴 淳一[3,d)]

受付日 2012年3月30日, 採録日 2012年9月10日

概要：RDF は多様な関係を表現できるグラフ構造データモデルであり，RDF で表現されたデータからは特徴的な関係情報を抽出することができる．我々は，RDF データのグラフ構造を解析し，関係を示すグラフパターンを抽出し，クエリとして利用する手法を検討している．RDF データを統合する場合，スキーマが冗長になる可能性がある．その場合，セマンティクスが類似するグラフパターンが多数発生するため，出現頻度などを用いた従来のフィルタリングでは，グラフパターンのバリエーションの確保ができなかった．この問題に対して，セマンティクスが類似するグラフパターンをクラスタリングする手法を提案する．また，本手法を企業内システムの実データに適用し，評価した結果を報告する．

キーワード：RDF，グラフマイニング，クラスタリング，データ統合，知識処理

A Proposal of Query's Graph Pattern Clustering for RDF Metadata Retrieval

Kyoji Iiduka[1,a)] Takahiko Murayama[1,b)] Toru Kobayashi[2,c)] Jun-ichi Akahani[3,d)]

Received: March 30, 2012, Accepted: September 10, 2012

Abstract: RDF is a labeled graph data format for representing various relations, and useful relationships can be extracted from RDF data. In order to use the relationships for constructing queries, we have been investigated methods that extract graph patterns from RDF data graph. When different RDF data is merged, some problems will be occur. One of these problems is that a lot of graph patterns with similar semantics are generated by duplication of RDF schema. Traditional occurrence rate filtering method cannot deal with RDF schema duplication, so resulting relationships do not have enough variation. This paper proposes a method that eliminates graph patterns with similar semantics based on clustering technique. We also show evaluation results by applying the method to real databases of an enterprise.

Keywords: RDF, graph mining, clustering, data integration, knowledge-processing

1. はじめに

近年，RDF（Resource Description Framework）化したデータを互いにリンク付けた Linked Data を公開する活動がさかんである [1]．英語版 Wikipedia の infobox 情報を RDF 化した DBpedia [2] を中心に，地理情報，メディア情報，科学論文，学術情報，政府情報など多種多様な情報のネットワークが形成されつつある．RDF は，人や物事などをリソースとし，リソース間の多様な関係を表現できるグラフ構造データモデルであり，リンクをたどることで，様々な関係を見出せる．RDF にはグラフパターンを

1 日本電信電話株式会社 NTT ソフトウェアイノベーションセンタ
NTT Software Innovation Center, NIPPON TELEGRAPH AND TELEPHONE CORPORATION, Musashino, Tokyo 180–8505, Japan
2 日本電信電話株式会社 NTT サービスエボリューション研究所
NTT Service Evolution Laboratories, NIPPON TELEGRAPH AND TELEPHONE CORPORATION, Yokosuka, Kanagawa 239–0847, Japan
3 NTT アドバンステクノロジ株式会社
NTT Advanced Technology Corporation, Kawasaki, Kanagawa 210–0007, Japan
a) iiduka.kyo@lab.ntt.co.jp
b) murayama.takahiko@lab.ntt.co.jp
c) kobayashi.toru@lab.ntt.co.jp
d) junichi.akahani@ntt-at.co.jp

クエリとして使用できる言語 SPARQL [3] が用意されている．様々な情報源のデータをつなげた RDF データからは，個々の情報源からでは探すことができない多様な関係性を探り出すことが可能となる．

2. RDF データの検索

企業内には，様々な業務を遂行するために，多種多様なデータが蓄積されている．これらのデータは，業務遂行のために個別最適化されているため，一般的に断片的な情報しか含まない．しかし，これらのデータを RDF に変換してつなげることで，断片的な情報からは得られない，企業内に潜在する知識を抽出できるようになる．たとえば，顧客から “セキュリティ” に関する技術的な問合せを受けた営業担当者が，企業内にいる “セキュリティ” に詳しい人物を探す場面を考えてみる．営業担当者は，“セキュリティ” の専門家がいそうな組織を人づてに探し，見つけた専門家が顧客の求める知識を持つ人物か確かめることになるだろう．このような作業を支援するために，ブログなどソーシャルメディアを用いて情報共有する試みがある [4]．しかし，業務と直結しない作業に対するインセンティブは低く，社員は積極的な情報提供をしない．これに対して，業務遂行上に蓄積されるデータを利活用できれば，社員からの自発的な情報提供に頼ることなく，社員が持つ様々な技能を検索できると考えられる．

そこで我々は，RDF 化した企業内情報から有用な知識の抽出を可能とするために，RDF データの部分グラフであるグラフパターン抽出手法を検討した．出現頻度の高いグラフパターンは有用なクエリと考え，RDF データのグラフ構造を解析してグラフパターンを抽出し，クエリとして利用する手法を提案した [5], [6]．しかし，実際に企業内情報を用いて出現頻度が高いグラフパターンを抽出したところ，セマンティクスが類似するグラフパターンが大量に抽出され，出現頻度のみによる抽出では十分なバリエーションを確保できないことが明らかになった [7], [8]．たとえば，「研究が所属する組織」や「研究の責任者が所属する組織」など，出現頻度が高くセマンティクスが類似するグラフパターンが多数存在すると，「技術の開示先組織」など出現頻度が低いグラフパターンを抽出できなくなる．このような現象が起こる原因は，個別最適化されたデータをつなげた結果，スキーマが冗長になり，類似するセマンティクスを示すが，表現が異なる関係が複数生じたためである．

セマンティクスが類似するグラフパターンを排除するためには，スキーマから冗長性をなくせばよい．そのためには各情報源のデータを精査し，欠損や冗長性がない新たなスキーマを設計する必要が出てくる．しかしこの作業は，つなげるデータの種類が多くなるほど煩雑になり，膨大な作業コストを必要とする．また，新たな種類のデータを追加した場合や外的要因などによりスキーマが変わる場合は，再び全体のスキーマを設計し直す必要がある．

このように，スキーマの設計変更は膨大な作業コストがかかる．そこで我々は，冗長性があるスキーマに則ったデータからグラフパターンを抽出し，セマンティクスが類似するグラフパターンに分類して，類似するグラフパターンを排除するアプローチをとる．我々の経験によると，企業内情報から抽出した 2 つのリソースをつなぐグラフパターンの中で，同一のリソースを多くつなぐグラフパターンは，セマンティクスが類似する傾向があった [8], [9]．そこで本論文では，「同一のリソースを多くつなぐグラフパターンは，セマンティクスが類似する」と定義し，グラフパターンを自動的に分類するクラスタリング手法を提案する．これにより，作業コストを抑えたうえで，グラフパターンのバリエーションを確保することが可能となる．

本論文では，2 章で RDF データからグラフパターンを抽出する際の課題をあげ，3 章でグラフパターンのクラスタリング手法を提案する．4 章で実データを用いた評価実験と結果を示し，5 章で考察をする．

2.1 RDF データのグラフ表現

RDF ではリソースおよびリソース間の関係を，サブジェクト（主語），プロパティ（述語），オブジェクト（目的語）からなるトリプルで記述する．そのため，RDF データはラベル付き有向グラフとして表現できる．サブジェクトとオブジェクトはリソースやリテラル値に該当し，グラフのノードとなる．プロパティはリソース間の関係に該当し，ノード間をつなぐアークとなる．

たとえば**図 1** は，研究組織における論文と研究に関する RDF データをグラフで表現したものである．クラス “人” のリソースには “山田太郎”，“田中二郎” と “情報三朗” があり，クラス “組織” のリソースには “X グループ” と “Y グループ” がある．クラス “人” とクラス “組織” の間には，プロパティ “所属” でつながる関係が表現されている．

2.2 グラフパターン

グラフパターンとは，サブジェクト，プロパティ，オブジェクトの一部が，任意の値をとる変数となるトリプルであり，SPARQL [3] などの RDF クエリ言語の中で，RDF データ内の複雑な関係を検索するための検索条件として使用する．

本論文では，任意の 2 つのクラスを特定し，それぞれのクラスに属するリソースをつなぐノードが変数となるグラフパターンを対象とする．特定した 2 つのクラスを端点クラスと呼び，端点クラスに属するリソースの変数を端点変数と呼ぶこととする．このようなグラフパターンをクエリに用いると，端点クラスに属するリソース間の関係を検索することができる．端点変数にマッチしたリソースを，端点リソースと呼ぶこととする．

図 2 に，図 1 の RDF データを検索するためのグラフパ

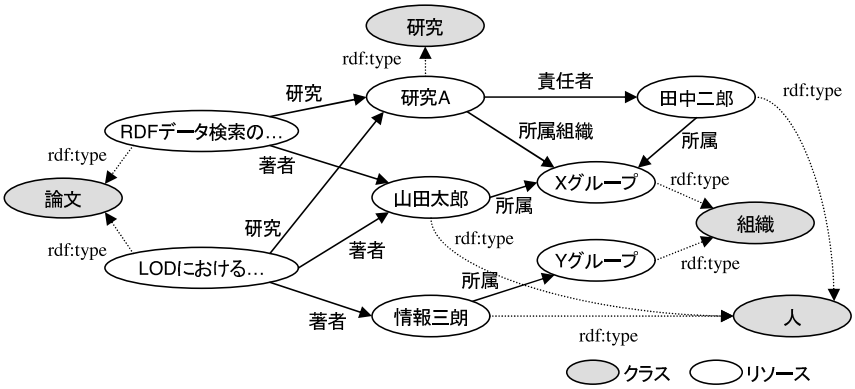


図 1 RDF データのグラフ表現
Fig. 1 RDF graph representation.

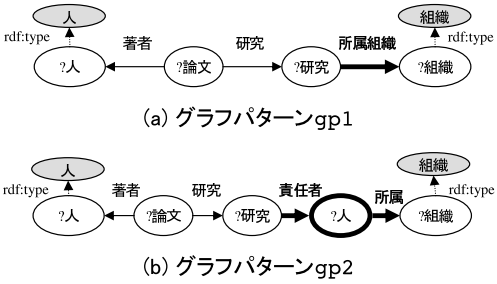


図 2 “人”, “組織” 間のグラフパターン例
Fig. 2 Graph patterns example from “person” to “organization”.

?人	?論文	?研究	?組織
山田太郎	RDFデータ検索の…	研究A	Xグループ
山田太郎	LODにおける…	研究A	Xグループ
情報三朗	LODにおける…	研究A	Xグループ

(a) gp1の検索結果

?人	?論文	?研究	?人	?組織
山田太郎	RDFデータ検索の…	研究A	田中二郎	Xグループ
山田太郎	LODにおける…	研究A	田中二郎	Xグループ
情報三朗	LODにおける…	研究A	田中二郎	Xグループ

(b) gp2の検索結果

図 3 検索結果
Fig. 3 Searching results.

ターンの例を示す．“?論文” や “?研究” など，“?” で始まる名前のノードは，任意のリソースに該当する変数とする．一方アークは，“著者” や “所属” などのプロパティの値である．図 2 (a)，(b) のグラフパターン gp1，gp2 ともに，端点クラスは “人” と “組織” であり，グラフパターンの両端にある変数 “?人” と “?組織” が端点変数となる．図 2 のグラフパターンは，人と組織の関係を示しており，gp1 は「ある人物が書いた論文の研究の所属組織」，gp2 は「ある人物が書いた論文の研究の責任者の所属する組織」と解釈できる．

このグラフパターンを，図 1 の RDF グラフに対するクエリとして用いて，得られた検索結果を**図 3** に示す．図 3 (a)，(b) ともに端点リソースは，{“山田太郎”, “情報三朗”} と {“X グループ”} になる．

2.3 グラフパターン抽出の要件

異なる情報源のデータをマージした RDF データは，類似するセマンティクスのクラスやプロパティが混在し，冗長なスキーマが生じてしまう．冗長なスキーマからグラフパターンを抽出すると，セマンティクスが類似するグラフパターンが多数発生してしまう．その結果，類似するグラフパターンに埋もれ，冗長性を持たない有用なグラフパターンの抽出が困難になり，グラフパターンのバリエーションを確保し難い．

以下に，セマンティクスが類似するグラフパターンの例をあげる．図 1 の RDF データは，研究組織における論文 DB，研究 DB，研究員 DB をつないで得たものである．図 2 の 2 つのグラフパターンは，図 1 の RDF データから抽出したものである．通常，“研究” と「“研究” の “責任者”」は，同一 “組織” に所属する．そのため，オリジナルの研究 DB には責任者の所属情報はない．しかし，研究員 DB をつないだことで，gp1 とセマンティクスがほぼ一致する研究の責任者を経由した gp2 も抽出される．

スキーマの冗長性を防ぐためには，両者をつなぐ共通スキーマを作成し，共通スキーマへ変換する必要がある．しかし，共通スキーマの作成と変換は一般的に難しく [10]，スキーマから冗長性を排除するコストは膨大になる．さらに，新たな情報の追加やデータのスキーマが変更された場合，そのつどこのコストが生じる．企業の場合，ビジネスルールの変更やシステムの更新によって，データスキーマの変更が頻繁に生じる可能性があり，共通スキーマへの変換は現実的な手法とはいえない．

このような状況において，グラフパターンのバリエーションを確保するため，以下の 3 つの要件を設定した．

(1) 類似するセマンティクスのグラフパターンを排除しつつ，グラフパターンのセマンティクスのバリエーションを確保できること
(2) スキーマ変更時のコストを低く抑えられること
(3) グラフパターン抽出処理の計算量が少ないこと

3. アプローチ

本章では，グラフパターンのセマンティクスの類似性を定義し，それに基づいて得られたクラスタから代表的なグラフパターンを抽出することでバリエーションを確保する手法を提案する．

まず，グラフパターンのセマンティクスの類似性を定義する．グラフパターンの意味論として SPARQL の意味論 [3] を用いると，「任意の RDF データに対し，グラフパターン p_1 と p_2 の端点リソース対が一致すれば，p_1 と p_2 のセマンティクスは一致する」ことを導き出せる．したがって，「ある RDF データに対し，p_1 と p_2 の端点リソース対集合が一致すれば，p_1 と p_2 のセマンティクスは一致する可能性がある」といえる．ここから，「ある RDF データに対して，p_1 と p_2 の端点リソース対集合の共通部分が多ければ，p_1 と p_2 のセマンティクスは類似する」といえる．

そこで，グラフパターンのセマンティクスの類似性を以下のように定義する．

定義 1 グラフパターンの端点リソース対の集合が類似する場合，グラフパターンのセマンティクスは類似する．

端点リソース対の集合の類似性は，端点リソース対の分布状況を特徴ベクトル化し，距離関数で定義する．

たとえば，図 3 を見ると，gp1 と gp2 の端点リソース対の集合は，{⟨“山田太郎”, “X グループ”⟩, ⟨“情報三朗”, “X グループ”⟩} となり一致する．一方，2.2 節で述べたように，gp1 は「ある人物が書いた論文の研究の所属組織」，gp2 は「ある人物が書いた論文の研究の責任者の所属する組織」を意味し，類似するセマンティクスを示している．

定義 1 より，セマンティクスが類似するグラフパターンにクラスタリングが可能となる．そこで，各クラスタから代表的なグラフパターンを抽出すれば，類似するセマンティクスのグラフパターンを排除できる．

次に，2.3 節の要件 (1)「類似するセマンティクスのグラフパターンを排除しつつ，グラフパターンのセマンティクスのバリエーションを確保」する方法を説明する．図 4 に，抽出グラフパターンの端点リソース対の概略を示す．まず，図 4 (A) のように，端点リソース対が少ない場合は，リソース対の網羅性が不十分で，バリエーションを確保しているとはいえない．また，図 4 (B) のように，端点リソース対の重複が多い場合は，類似するセマンティクスのグラフパターンが排除されていない．以上より，図 4 (C) のように，リソース対を網羅し，端点リソース対の重複を少なくできれば，要件 (1) を満たすことができる．

3.1 グラフパターンのクラスタリング

以下，グラフパターンのクラスタリング手法を述べる．

まず，検索対象の RDF データと 2 つの端点クラスを定め，以下の 4 段階の処理を行うことによりグラフパターンのクラスタリングを行う．

処理 1 グラフパターンの抽出
RDF データから端点クラスをつなぐ複数のグラフパターンを抽出する．

処理 2 リソース対の選定
端点クラスに属するリソースの対を選定する．
この処理は，特徴ベクトル作成処理における探索範囲を狭め，計算量を抑えるために行う．

処理 3 特徴ベクトル作成
処理 1 で抽出したグラフパターンに対して，処理 2 で抽出したリソース対に含まれる端点リソース対集合を抽出し，端点リソース対集合からグラフパターンの特徴ベクトルを作成する．

処理 4 クラスタリング
処理 3 で作成した特徴ベクトルを用いて，グラフパターンをクラスタリングする．

以下に，各処理の詳細を示す．

処理 1 グラフパターンの抽出

RDF データからスキーマ構造グラフを抽出し，2 つの端点クラスを含む部分グラフを抽出することで実現する．部分グラフの抽出には，幅優先探索，深さ優先探索を用いたシンプルな手法のほか，頻出する部分グラフを抽出する手

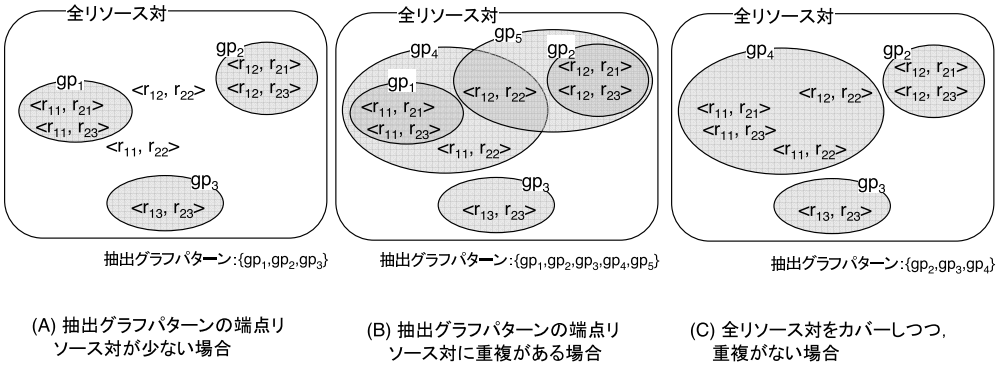


図 4 抽出グラフパターンの端点リソース対
Fig. 4 Extracted graph pattern's resource pairs.

法 [11]，Greedy 探索を行う手法 [12]，ランダムウォークによる経路探索を用いた手法 [14], [15], [16] など，様々な手法が提案されている．グラフパターンの抽出処理そのものは本論文の検証範囲でないため，今回は，探索範囲の制限を加えた幅優先探索を用いて抽出することとする．

処理 2　リソース対の選定

処理 3 の特徴ベクトル作成処理は，全リソース対を端点変数に代入してマッチングを行う計算量の多い処理であるため，リソース対を削減して計算量を抑える．ただし，選定数が少ないと端点リソース対の分布状況の差異が得られず，クラスタリング精度が下がるために考慮が必要である．

そこで，様々なグラフパターンの端点リソース対となりうる，多くのリソースに接続するリソースどうしの対を候補として選び出すこととする．その際，プロパティの偏りが生じないように，プロパティごとに接続数をカウントし，接続数が上位のものを選び出す．

たとえば，図 1 の RDF から，端点クラス ⟨“人”, “組織”⟩ のリソース対を選び出す場合を考える．“人” のリソースは，“著者”，“責任者” のプロパティで接続する．それぞれのプロパティで接続数が最も多いリソースを選ぶとする．プロパティ “著者” での接続数が多いリソースは “山田太郎”，プロパティ “責任者” での接続数が多いリソースは “田中二郎” となり，“山田太郎” と “田中二郎” の 2 つのリソースが選ばれる．“組織” についても同様にリソースを選び，選ばれたリソースどうしの対を生成すれば，処理は完了する．

処理 3　特徴ベクトル作成

各グラフパターンの端点リソース対の分布状況をベクトル化し，これを特徴ベクトルとしてクラスタリングを行う．グラフパターンの特徴ベクトルは，処理 2 で得た全リソース対を座標とし，値に $\{0,1\}$ を持つ多次元ベクトルとする．各座標の値は，各リソース対が端点リソース対となる場合は 1 とし，端点リソース対ではない場合は 0 とする．

以下に，図 2 のグラフパターン gp1 を例に説明する．処理 2 において，{⟨“山田太郎”, “X グループ”⟩, ⟨“山田太郎”, “Y グループ”⟩, ⟨“田中二郎”, “X グループ”⟩, ⟨“田中二郎”, “Y グループ”⟩} の 4 組のリソース対が選定されたとする．リソース対 ⟨“山田太郎”, “X グループ”⟩ は，gp1 の端点リソース対であるため，値は 1 となる．一方，リソース対 ⟨“山田太郎”, “Y グループ”⟩, ⟨“田中二郎”, “X グループ”⟩, ⟨“田中二郎”, “Y グループ”⟩ は，gp1 の端点リソース対ではないため，値は 0 となる．その結果，特徴ベクトルは $(1,0,0,0)$ となる．同様に，gp2 の特徴ベクトルも $(1,0,0,0)$ となる．

処理 4　クラスタリング

クラスタリングアルゴリズムには，近似データどうしを融合させ樹形図を作成する階層クラスタリングと，分割と評価関数の再計算を繰り返し，最適な評価値を持つクラスタに分割する非階層クラスタリングがある [13]．今回は，適切なクラスタの分割数を決定することができないため，クラスタ分割数の調整が容易な，階層クラスタリングアルゴリズムを用いる．

3.2　代表的なグラフパターンの抽出

次に，適切なクラスタ数を決定した後，各クラスタの中から代表元を抽出する．代表元は，対象となるクラスタの中で最も多くのリソース対にマッチするグラフパターンとする．そこで問題となるのは，クラスタ数の決定方法である．

3 章で述べたように，2.3 節の要件 (1)「類似するセマンティクスのグラフパターンを排除しつつグラフパターンのセマンティクスのバリエーションを確保」するためには，以下を満たす必要がある．

(i) 抽出したグラフパターンの端点リソース対が，リソース対を網羅すること
(ii) 端点リソース対の重複を少なくすること

そこで，クラスタ数を決定するために，以下の 2 つの指標を用いることとする．

クラスタ数 n のカバー率：

$$C_n = \frac{|M(R_n)|}{|M|} \quad (0 \le C_n \le 1) \tag{1}$$

クラスタ数 n のユニークカバー率：

$$uC_n = \frac{|uM(R_n)|}{|M|} \quad (0 \le uC_n \le 1) \tag{2}$$

R_n はクラスタ数 n のクラスタリングで得られた代表元の集合とする．M と $M(R_n)$ は，$M(p)$ をグラフパターン p の端点リソース対の集合とした場合，$M = \bigcup_{p \in All} M(p)$，$M(R_n) = \bigcup_{p \in R_n} M(p)$ とする．また，$uM(R_n)$ は，R_n に属するグラフパターンの端点リソース対のうち，他のグラフパターンの端点リソース対にならないものの和集合とする．

C_n は，全リソース対に対する代表元の端点リソース対の和集合の割合を示す．C_n の値が大きくなればリソース対の網羅率が高くなり，上記 (i) を満たすことになる．一方，uC_n は，ただ 1 つの代表元の端点リソース対となるリソース対の集合の割合を示す．uC_n の値が大きくなれば端点リソース対の重複が少なくなり，上記 (ii) を満たすことになる．したがって，複数のクラスタ数 n の C_n と uC_n を比較し，ともに高い数値を示すクラスタ数を選択すれば，代表元として得られるグラフパターンは，少数でバリエーションを確保したものとなる．

C_n と uC_n はトレードオフの関係となるため，ともに高い値を示すクラスタ数は存在しない．そこで，C_n と uC_n の調和平均が高いクラスタ数を最適なものとする．調和平均の算出方法を，以下の数式で示す．

調和平均：

$$H_{\beta,n} = \frac{(1+\beta^2)\cdot \overline{C}_n \cdot \overline{uC}_n}{\overline{C}_n + \beta^2 \cdot \overline{uC}_n} \quad (\beta \geq 0, 0 \leq H_{\beta,n} \leq 1) \tag{3}$$

$\overline{C}_n$ は正規化したカバー率 C_n で，$\overline{C}_n = \frac{C_n - min_k C_k}{max_k C_k - min_k C_k}$ とし，0 から 1 までの値をとる．$\overline{uC}_n$ は正規化したユニークカバー率 uC_n で，$\overline{uC}_n = \frac{uC_n - min_k uC_k}{max_k uC_k - min_k uC_k}$ とし，0 から 1 までの値をとる．β は $\overline{C}_n$ に対する重みで，$\beta \geq 1$ の場合カバー率を重視した値となる．バリエーションの確保を優先させるために $\beta = 2$ とする．

4. 実験

我々が所属する研究所内で業務管理に利用されている，研究員 DB，論文 DB，研究 DB，研究成果 DB に関する 7 種類の実データを，RDF に変換して評価実験を行った．各データは，RDB ないしは Excel ファイルとして管理されている構造化データである．

RDF 化に際しては，基本的にオリジナルのデータスキーマをそのまま再現した．ただし，他のデータセットとの接点を増やすため，多少スキーマの修正を加えた．その結果，11 種類の RDF クラス，リソース間をつなぐ 33 種類のプロパティで構成された RDF データとなった．使用したデータセットの概要を**表 1** に示す．

4.1 実験手順

実験は，**表 2** に示す検索ニーズが高い 8 組の端点クラス対のグラフパターンを対象とする．

表 1 使用データの概要
Table 1 Summary of used dataset.

データ名	データ規模（概算）
研究員 DB	2.5 万トリプル
論文 DB	16 万トリプル
研究 DB	2.4 万トリプル
研究成果情報 DB	0.5 万トリプル
成果利用情報 DB	1 万トリプル
技術支援情報 DB	1.2 万トリプル
技術開示情報 DB	0.1 万トリプル
計	23.7 万トリプル

表 2 使用グラフパターンの概要
Table 2 Summary of used graph patterns.

端点クラス対	グラフパターン数	端点リソース対数
⟨“組織”,“組織”⟩	103 個	3,049
⟨“組織”,“研究”⟩	103 個	2,437
⟨“組織”,“技術”⟩	44 個	1,225
⟨“人”,“組織”⟩	117 個	4,217
⟨“人”,“人”⟩	133 個	11,017
⟨“人”,“研究”⟩	108 個	4,590
⟨“人”,“技術”⟩	46 個	6,430

まず，グラフパターンを抽出する（**処理 1**）．今回は，以下の条件を満たすグラフパターンをすべて抽出して使用した．

- 端点クラス間を 1 本のパスでつなぐもの
- パターンを構成する変数ノードの数が 5 以下
- アークの向きが変わる変数ノードの数が 1 つ以下

次に，端点クラスとなる “組織”，“人”，“研究”，“技術” のリソースを選定する（**処理 2**）．今回は，プロパティごとに上位 20 個を抽出した．抽出したリソースを組み合わせ，端点リソース対の集合を生成する．以上の条件のもとで抽出したグラフパターン数と，端点リソース対数を表 2 に示す．

さらに，グラフパターンの特徴ベクトルを作成し（**処理 3**），クラスタリングを行う（**処理 4**）．クラスタリングアルゴリズムは，分類精度が高いとされている Ward 法 [28] を使用する．

クラスタ数は，クラスタ数が 2 個から 40 個までの 20 セットのクラスタ群を抽出した後，3.2 節で定義した $\overline{C}_n$ と $\overline{uC}_n$ を比較して決定する．

4.2 実験結果

4.2.1 クラスタ数

クラスタ数に対する正規化カバー率 $\overline{C}_n$ と正規化ユニークカーバ率 $\overline{uC}_n$，および $\overline{C}_n$ と $\overline{uC}_n$ の調和平均値 $H_{2,n}$ の結果を示す．

クラスタ数を 2 個から 40 個まで増やすと，代表元となるグラフパターンの数が多くなるため，$\overline{C}_n$ は 1.0 に近くなる．その反面，$\overline{uC}_n$ は低下する．**図 5** に，クラスタ数ごとの端点クラス対 ⟨“人”, “組織”⟩ と ⟨“人”, “技術”⟩ の $\overline{C}_n$ と $\overline{uC}_n$ および $H_{2,n}$ の推移を示す．

3.2 節で述べたとおり，$H_{2,n}$ が最も高い値を示すクラスタ数に決定する．上記基準を適用して決定したクラスタ数を**表 3** に示す．

4.2.2 グラフパターン抽出結果

4.2.1 項で決定したクラスタ数を用いて，グラフパターンのクラスタリングと代表元抽出を行った．

評価にあたり，グラフパターン p のカバー率 $C(p)$ を使用する．$C(p)$ は，全グラフパターンのリソース対 M に対する p のリソース対 $M(p)$ の割合とし，0 から 1 までの値

表 3 クラスタ数
Table 3 Culuster number.

端点クラス対	クラスタ数
⟨“組織”,“組織”⟩	8
⟨“組織”,“研究”⟩	16
⟨“組織”,“技術”⟩	8
⟨“人”,“組織”⟩	10
⟨“人”,“人”⟩	6
⟨“人”,“研究”⟩	8
⟨“人”,“技術”⟩	6

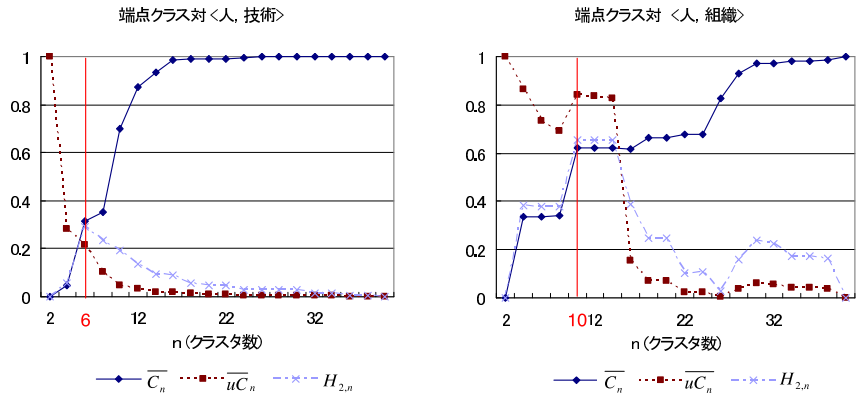


図 5 $\overline{C}_n$，$\overline{uC}_n$，および $H_{2,n}$ の推移
Fig. 5 $\overline{C}_n$, $\overline{uC}_n$ and $H_{2,n}$.

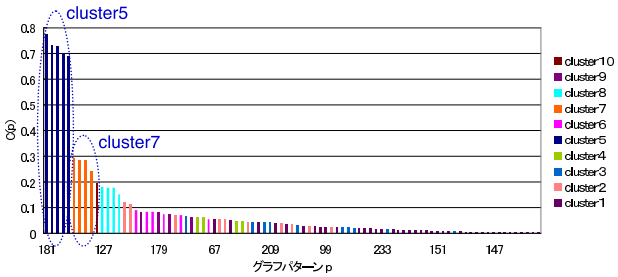


図 6 端点クラス対 ⟨“人”, “組織”⟩ のグラフパターンのクラスタ
Fig. 6 Clusters of ⟨“person”, “organization”⟩ graph pattern.

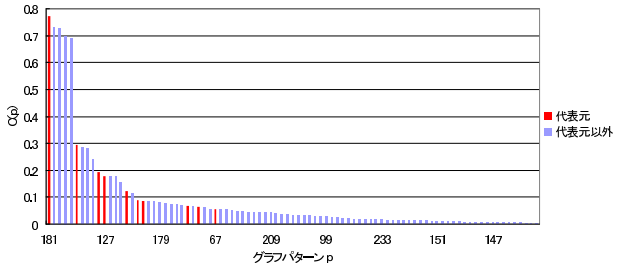


図 7 端点クラス対 ⟨“人”, “組織”⟩ から抽出した代表元
Fig. 7 Representatives of ⟨“person”,“organization”⟩ graph pattern.

をとる．$C(p)$ は，RDF データ内での p の出現頻度に比例する値である．

図 6 と**図 7** に端点クラス対が ⟨“人”, “組織”⟩ のグラフパターンに対する $C(p)$ をプロットした結果を示す．図 6，図 7 ともに，縦軸に $C(p)$ を，横軸に $C(p)$ が高い順にソートした p の ID をプロットしている．図 6 では，同じクラスタに属するグラフパターンは同じ色で表示している．図 7 は，各クラスタから最も $C(p)$ が高い p を代表元として選び，代表元を赤色にプロットしたものである．

結果を見ると，同一クラスタに属する p の $C(p)$ が近い様子が確認できた．たとえば図 6 を見ると，*cluster5* に属するグラフパターン p_{c5} のカバー率 $C(p_{c5})$ は 0.8 から 0.7 の間に固まっており，*cluster7* に属するグラフパターン p_{c7} のカバー率 $C(p_{c7})$ は 0.3 から 0.25 の間にあり，上位を独占している様子が確認できる．このような $C(p)$ の分布に対して，従来の出現頻度のみによるグラフパターンの抽出では，同一クラスタに属するグラフパターンだけが抽出されてしまう．

一方，クラスタリングの結果得られる代表元は，$C(p)$ の値が分散していることが確認できた．たとえば図 7 を見ると，$C(p)$ の値が低い p が代表元として抽出されている様子が確認できる．

5. 考察

本章では，提案した手法が，2.3 節で示した 3 つの要件を満足するか考察する．

5.1 グラフパターンのセマンティクスのバリエーション

2.3 節の要件で，類似するセマンティクスのグラフパターンを排除しつつ，グラフパターンのセマンティクスのバリエーションを確保できることをあげた．本節では，出現頻度を用いたグラフパターン抽出手法と比較し，この要件を満足するか検証する．

検証は以下のように行う．まず，データに付随する業務の背景知識を用いて，グラフパターンのセマンティクスを調査する．次に，既存手法と提案手法で抽出したグラフパターンのうち，上記セマンティクスの抽出具合を，F 値および F_2 値で評価を行う．

まず，背景知識を用いたセマンティクスの調査について説明する．データに付随する業務の背景知識を用いると，類似する概念や関係を見出すことができる．そこで，類似概念や関係を同一概念や関係に置き換える縮約ルールを手動で設定する．この縮約ルールを適用した結果，同じ形に縮約するグラフパターンを，縮約ルールを用いた類似セマンティクスとする．今回の考察のために，使用データに関する業務の背景知識から 12 個の縮約ルールを抽出した．**図 8** に，縮約ルールの一部を例示する．

たとえば**図 9** は，端点クラス対が ⟨“人”, “組織”⟩ のグラフパターンと，図 8 の縮約ルールを適用して得られるグラ

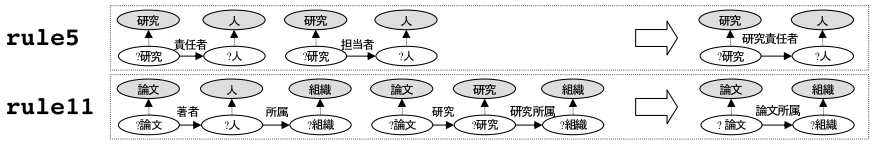


図 8 縮約ルールの例
Fig. 8 Example of contraction rules.

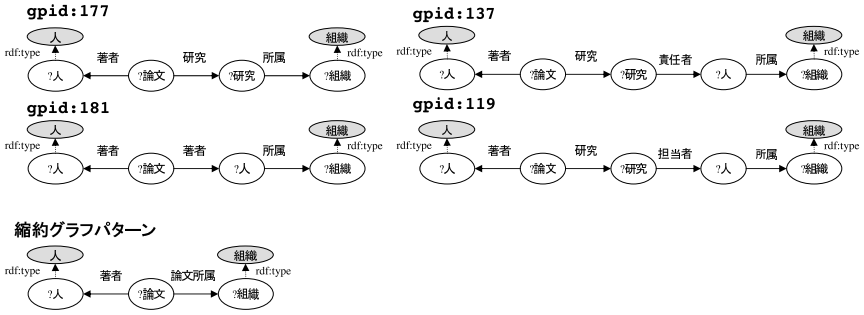


図 9 縮約ルールを用いた類似セマンティクスのグラフパターン
Fig. 9 Graph patterns of similar semantics.

表 4 抽出セマンティクス数
Table 4 Number of the extracted semantics.

端点クラス対	全セマンティクス数	提案手法	既存手法 上位 5 個	既存手法 上位 15 個	$C(p)$ 尖度
⟨“組織”, “組織”⟩	15	**7(47%)**	3(20%)	4(27%)	1.43
⟨“組織”, “研究”⟩	7	5(71%)	2(29%)	5(71%)	6.73
⟨“組織”, “技術”⟩	5	**4(80%)**	2(40%)	3(60%)	−1.43
⟨“人”, “組織”⟩	10	**5(50%)**	1(10%)	3(30%)	13.10
⟨“人”, “人”⟩	9	5(56%)	4(44%)	6(67%)	46.56
⟨“人”, “研究”⟩	6	4(67%)	3(50%)	4(67%)	17.60
⟨“人”, “技術”⟩	4	3(75%)	3(75%)	3(75%)	25.05

フパターンである．gpid:177 は「ある人物が書いた論文が属する研究の所属組織」，gpid:181 は「ある人物が書いた論文の著者の所属組織」，gpid:137 は「ある人物が書いた論文が属する研究の責任者の所属組織」とあり，“論文” や “研究” などのリソースを経由する複雑な関係に見える．しかし，gpid:137 と gpid:119 は，図 8 の rule5 を適用すると gpid:177 と同じ形に縮約される．さらに，gpid:177 と gpid:181 は，図 8 の rule11 を適用すると図 9 の縮約グラフパターンと同じ形に縮約される．したがって，図 9 の 4 つのグラフパターン gpid:177, gpid:181, gpid:137, gpid:119 は，縮約ルールを用いた類似セマンティクスである．

この縮約ルールを用いた類似セマンティクスの定義を用いて，4 章の評価実験で得られた代表元のバリエーションを検証する．

表 4 に，抽出した代表元の中で異なるセマンティクスの数をカウントした結果を示す．比較のため，出現頻度で抽出した既存手法による代表元の結果も示す．提案手法で得た代表元の個数が 6 個から 18 個あるため，提案手法と条件が近い上位 5 個から 15 個の代表元を抽出する手法と比較を行う．たとえば，端点クラス対 ⟨“組織”, “技術”⟩ の全グラフパターン 44 個のうち，セマンティクスが異なるものは 5 種類が存在した．これに対し，提案手法で抽出した代表元には 4 種類のセマンティクスを含んでおり，80%のセマンティクスをカバーしている．一方既存手法では，出現頻度の高い上位 5 個を代表元として抽出する場合は 40%，上位 15 個を抽出する場合は 60%のセマンティクスをカバーするにすぎない．同様に，端点クラス対 ⟨“組織”, “組織”⟩ において既存手法で 20%から 27%のセマンティクスをカバーするのに対して提案手法で 47%，端点クラス対 ⟨“人”, “組織”⟩ において既存手法で 10%から 30%に対して提案手法では 50%のセマンティクスをカバーしており，既存手法よりセマンティクスのバリエーションが確保できることを確認した．

また，表 4 から，端点クラス対ごとに既存手法との有意性の違いが確認できる．⟨“組織”, “組織”⟩，⟨“組織”, “研究”⟩，⟨“組織”, “技術”⟩，⟨“人”, “組織”⟩ では既存手法に比べてバリエーションを確保している．一方，⟨“人”, “人”⟩，⟨“人”, “研究”⟩，⟨“人”, “技術”⟩ では既存手法との有意性はない．そこで，各端点クラス対の $C(p)$ の分布を分析した．その結果，既存手法との差がない 3 つの端点クラス対

表 5 セマンティクスの再現率と適合率
Table 5 Recal and precision of semantics.

平均値	提案手法	既存手法 上位 5 個	既存手法 上位 15 個	既存手法 上位 30 個	既存手法 上位 60 個	既存手法 上位 90 個
再現率	0.64	0.38	0.57	0.66	0.75	0.91
適合率	0.39	0.51	0.27	0.16	0.10	0.09
F 値	0.46	0.42	0.35	0.25	0.17	0.17
F_2 値	0.54	0.39	0.47	0.39	0.31	0.32

では，$C(p)$ の値が 1 付近ないしは 0 付近に 2 極化していた．これは，$C(p)$ の分布における尖度[*1]（表 4 の $C(p)$ 尖度）からも確認できる．$C(p)$ の値が 2 極化しているためにクラスタリングの精度が出ず，既存手法との有意差がなくなったと考えられる．

以上の考察から，$C(p)$ の分布における尖度が低い端点クラス対に対しては，既存手法よりセマンティクスのバリエーションが確保できることを確認した．

表 5 に，セマンティクスの再現率と適合率を示す．再現率が高ければセマンティクスのバリエーションが確保でき，適合率が高ければ類似するセマンティクスのグラフパターンが排除されていることになる．表 5 を見ると，提案手法の再現率は 0.64 と，既存手法における上位 30 個のグラフパターンを抽出した手法と同程度の再現率を示した．一方，適合率は 0.39 と，既存手法に比べて著しく高い値を示している．F 値で比較すると，提案手法は 0.46 で，既存手法における最も高い値を示す値に比べて上回っていることを確認した．さらに，再現率を重視した F_2 値[*2]で比較すると，提案手法は 0.54 と，既存手法における値を上回ることを確認した．

なお，今回評価対象としたセマンティクス数は計 56 個と少ない．しかし，セマンティクス数はリソース間の関係性の数であるため，容易に増やすことができない．今回使用したデータは，4 章で述べたように，11 種類の RDF クラスと 33 種類のリソース間を結ぶプロパティで構成された複雑なスキーマ構造のデータであり，表 2 にあるとおり，4.1 節に示すグラフパターン抽出条件で 44〜133 個のグラフパターンを抽出できる，入手可能な最良なデータセットである．したがって，評価対象のセマンティクス数が少ないものの，評価の妥当性はあると考えている．

以上の分析より，既存手法に比べて提案手法は，類似するセマンティクスのグラフパターンを排除しつつセマンティクスのバリエーションを確保でき，2.3 節の要件を満たしているといえる．

[*1] 分布の尖り具合を表す．
$Kurt = \left\{ \frac{n(n+1)}{(n-1)(n-2)(n-3)} \sum_{i=1}^{n} \left(\frac{x_i - \overline{x}}{s} \right)^4 \right\} - \frac{3(n-1)^2}{(n-2)(n-3)}$．
$Kurt = 0$ ならば，正規分布と同程度．$Kurt > 0$ ならば，正規分布よりとがっている．

[*2] $F_\beta = \frac{(1+\beta^2) \cdot precision \cdot recall}{(\beta^2 \cdot precision + recall)}$．$\beta = 1$ の場合，通常の F 値となる．$\beta > 1$ の場合，再現率を重視した値となる．

5.2 スキーマ変更時のコスト

2.3 節の要件で，スキーマ変更にともなうコストを低く抑えられることをあげた．

提案手法は，RDF データのスキーマ構造によらず適応できるアルゴリズムを用いているため，アルゴリズムの実行においてスキーマ変更にともなう調整は必要ない．そのため，データのスキーマ変更が生じた場合，新たなスキーマのデータを用いてグラフパターン抽出アルゴリズムを実行するだけで済む．その際に，アルゴリズム実行のためのマシン処理のコストだけで，人的な稼動は最小に抑えることができる．そのため，2.3 節の要件を満たしているといえる．

5.3 グラフパターン抽出処理の計算量

一般的にグラフデータの分析では，組合せ爆発が生じるため計算量が膨大になりがちになる．たとえば，N 個のリソースで構成されている RDF データに対して，変数が n 個のグラフパターンの端点リソース対を抽出する計算量は，最大 $O(N^n)$ となる．そのため，2.3 節では，グラフパターン抽出処理の計算量を低く抑えることを要件にあげた．

本手法では，計算量を抑えるため，グラフパターンのクラスタリングにおいて，端点リソースの選定を行っている．その結果，計算量は $O(N) + O(N^{n-2}) = O(N^{n-2})$ に収めることができる．

$O(N^{n-2})$ でも計算量が大きいが，以下の理由で現実的な計算量で抑えることができる．我々の経験からは，6 個以上のノード数のグラフパターンになると，その意味を解釈することが著しく困難となる．ただし，人が理解できないような複雑なグラフパターンを基にしたクエリにはニーズは少ないため，除外してもよいと考えられる．

また，企業内の DB から作られた RDF データでは，“論文” や “支援” などのノード次数が正規分布するクラス（正規分布クラス）と，“人” や “組織” などのノード次数が対数分布するクラス（次数分布クラス）の，2 種類のクラスがあることを確認している．正規分布クラスの次数はたかだか定数で抑えられるため，正規分布クラスの計算量は無視してかまわない．

多くのグラフパターンでは，正規分布クラスに属する変数を経由して，次数分布クラスに属する変数がつながる形式をとる．たとえば，図 9 の `gpid:181` のグラフパ

ターンの場合，正規分布クラスの “?論文” と次数分布クラスの “?人” を経由して端点リソースがつながる．その結果 gpid:181 の計算量は，$O(N)$ にとどまる．仮に，変数の数が 6 個で，端点変数以外の 1 つが正規分布クラスの変数の場合，最大でも $O(N^2)$ の計算量にとどまる．

したがって，企業内データでの検索用クエリを抽出する用途においては，十分に現実的な計算量に抑えられるといえる．

5.4 既存研究

RDF グラフからリソース間の関係を抽出する手法はいくつか提案されている．

グラフデータからリソース間の関係を抽出する手法には，Web ページのリンク解析に用いられる PageRank [17] や HITS [18]，SALSA [19] などがある．これらの手法を RDF グラフへ拡張した手法も提案されている [20], [21]．これらの手法は，リソース間の関係の傾向は把握できるが，グラフパターンのようなリソースを特定しない関係情報を抽出することはできない．

リンクの重要度を計算する手法に，ObjectRank [22] や SemRank [23] などがある．この手法は，あらかじめプロパティに重みを割り当てる必要があり，そのための技術やノウハウが必要となる．また，リンク解析を用いた手法は，グラフデータ全体を探索するため，膨大な計算量を必要とする．RDF は，多様な関係を記述できるため，グラフデータの規模が大きくなりやすく，実用的な計算量を超えてしまう．

スキーマグラフを探索する手法は，スキーマグラフからグラフパターンを抽出した後，ランキングやフィルタリングする手法である．グラフパターンは，スキーマ構造から抽出できるため，グラフデータを探索する手法に比べて，計算量を抑えることができる．最もシンプルな手法として，ノード数を制限した 1 本パスのグラフパターンを抽出するものがある [24]．この手法は，グラフパターンをパス長以外に評価しないため，多数のグラフパターンが抽出されてしまう．そのため，グラフパターンを評価し，ランキングやフィルタリングをする必要が出てくる．

ユーザのニーズに合わせて，グラフパターンをランキングする手法がある [25]．たとえば，考慮したいアークを強調し，不要なアークを排除することで，ユーザの要求に応じたグラフパターンを抽出する．この手法は，事前にユーザが関心あるドメインに属するアークを選別して重み付けする必要があり，データに精通していることが必要である．

グラフパターンがグラフデータ中に出現する頻度により，グラフパターンの有用性を評価してフィルタリングする手法がある [5], [6], [26]．しかし，この手法を実データに適用すると，セマンティクスが類似するパターンが多く抽出され，バリエーションの確保ができないことが明らかになった [7]．

類似するパターンを集約する方法に，RDF のオントロジ階層情報を用いて，類似パターンを集約する手法がある [27]．この手法は，多重階層の複雑なオントロジを持つ RDF でのみ効果が発揮される．しかし，複雑なオントロジはメンテナンスコストが高く，一般的に用いられていない．

既存手法は，ユーザの嗜好や優先度などデータ自体から得られない情報を利用する手法を除くと，出現頻度からグラフパターンを抽出する手法に集約される．そこで，5.1 節において提案手法と出現頻度を用いた手法との比較を行った．その結果，セマンティクスの異なるグラフパターンの抽出において，F_2 値が既存手法で 0.39 に対して，提案手法が 0.54 と高い値を示した．これにより，既存手法に比べて多様なセマンティクスのグラフパターンが抽出できることを確認した．

なお，企業内情報以外においても，人や組織など普遍的な概念を示すクラスのリソースは，スモールワールド性 [30] を示す可能性が高い．一方，普遍的な概念間をつなぐリソースは，個々のイベント的に発生する事象の可能性が高く，正規分布，ないしは類する分布を示すと思われる．したがって，企業内情報以外のデータに対しても，今回の知見の展開が可能と考えられる．

6. まとめ

異なる情報源のデータをマージした RDF グラフからグラフパターンを抽出する場合，スキーマが冗長になる可能性があり，類似するセマンティクスのグラフパターンが多数発生するという問題を指摘した．この問題に対し，セマンティクスが類似するグラフパターンごとにクラスタリングし，代表的なグラフパターンを抽出する手法を提案して，実データを用いた評価実験を行った．実験の結果，同じクラスタに分類されるグラフパターンの出現頻度が類似していることが確認でき，共通スキーマの作成などのコストを抑え，従来手法で確保できないセマンティクスが類似する冗長なグラフパターンを排除しつつバリエーションを確保できることを確認した．

参考文献

[1] 長野伸一，萩野達哉ほか：リンクするデータ（Linked Data）—広がり始めたデータのクラウド，情報処理，Vol.52, No.3, pp.282–333 (2011).
[2] DBpedia: Wiki.dbpedia.org (online), available from ⟨http://wiki.dbpedia.org/⟩.
[3] W3C: SPARQL Query Language for RDF, W3C Recommendation (online), available from ⟨http://www.w3.org/TR/rdf-sparql-query/⟩.
[4] 和田　恭：米国におけるソーシャルメディアのビジネス利用に関する動向，IPA ニューヨークだより (online)，入手先 ⟨http://www.ipa.go.jp/about/NYreport/201107.pdf⟩.
[5] Sato, H., Iiduka, K., Mukaigaito, T. and Murayama, T.: Finding Similarity and Comparability from Merged Hetero Data of the Semantic Web by Using Graph Pattern

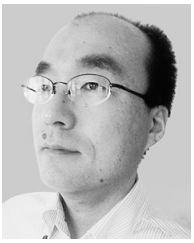
Matching, *WWW2005 Workshop, Activities on Semantic Web Technologies in Japan* (2005).
[6] 飯塚京士, 佐藤宏之, イコプラムディオノ, 村山隆彦：RDF データを対象としたグラフ検索におけるクエリ生成方式の検討，人工知能学会，SIG-SWO-A502-08 (2005).
[7] 酒井理江, 飯塚京士, 佐藤宏之, 村山隆彦, 小林 透, 服部宏充，石田 亨：Linked Data から潜在的な関係を探すためのクエリグラフパターン最適化，情報処理学会論文誌，Vol.51, No.12, pp.2298–2309 (2010).
[8] 飯塚京士，山本具英，大友健治，村山隆彦：RDF グラフ検索におけるクエリ類似性判定手法，情報科学技術フォーラム講演論文集，Vol.8, No.2, pp.507–508 (2009).
[9] 山本具英, 飯塚京士, 大友健治, 村山隆彦：RDF グラフ検索における部分パターンの情報量に着目したクエリ判定方法，情報科学技術フォーラム講演論文集，Vol.8, No.2, pp.475–476 (2009).
[10] Halevy, A.Y.: Why Your Data Won't Mix: Semantic Heterogeneity, *ACM Queue*, pp.50–58 (2005).
[11] Inokuchi, A., Washio, T. and Motoda, H.: An Apriori-Based Algorithm for Mining Frequent Substructures from Graph Data, *Proc. 4th PKDD*, pp.13–23 (2000).
[12] Nguyen, P., Ohara, K., Motoda, H. and Washio, T.: Cl-GBI: A Novel Approach for Extracting Typical Patterns from Graph-Structured Data, *Proc. PAKDD 2005*, pp.639–649 (2005).
[13] Hartigan, J.A.: *Clustering Algorithms*, John Wiley and Sons Inc. (1975).
[14] Faloutsos, C., McCurley, K. and Tomkins, A.: Fast discovery of connection subgraphs, *Proc. 10th SIGKDD*, pp.118–127, ACM (2004).
[15] Tong, H. and Faloutsos, C.: Center-Piece Subgraphs: Problem Definition and Fast Solutions, *Proc. 12th SIGKDD*, pp.404–413, ACM (2006).
[16] Koren, Y., North, S.C. and Volinsky, C.: Measuring and extracting proximity in networks, *Proc. 12th SIGKDD*, pp.245–255, ACM (2006).
[17] Brin, S. and Page, L.: The anatomy of a large-scale hypertextual web search engine, *Computer Networks and ISDN Systems*, Vol.30, No.1-7, pp.107–117 (1998).
[18] Kleinberg, J.M.: Authoritative sources in a hyperlinked environment, *J. ACM*, Vol.46, No.5, pp.604–632 (1999).
[19] Lempel, R. and Moran, S.: SALSA: The Stochastic Approach for Link-Structure Analysis, *ACM Trans. Information Systems*, Vol.19, No.2, pp.131–160 (2001).
[20] Kolda, T.G., Bader, B.W. and Kenny, J.P.: Higher-Order Web Link Analysis Using Multilinear Algebra, *Proc. 5th ICDM*, pp.242–249 (2005).
[21] Franz, T., Schultz, A., Sizov, S. and Staab, S.: Triplerank: Ranking semantic web data by tensor decomposition, *Proc. 8th ISWC*, pp.213–228 (2009).
[22] Balmin, A., Hristidis, V. and Papakonstantinou, Y.: Objectrank: Authority-based keyword search in databases, *Proc. 30th VLDB*, pp.564–575 (2004).
[23] Anyanwu, K., Maduko, A. and Sheth, A.P.: SemRank: Ranking complex relationship search results on the semantic web, *Proc. 14th WWW*, pp.117–127 (2005).
[24] Auer, S. and Lehmann, J.: What Have Innsbruck and Leipzig in Common? Extracting Semantics from Wiki Content, *Proc. 4th ESWC*, pp.503–517 (2007).
[25] Aleman-Meza, B., Halaschek-Wiener, C., Arpinar, I.B., Ramakrishnan, C. and Sheth, A.P.: Ranking complex relationships on the semantic web, *IEEE Internet Computing*, Vol.9, No.33, pp.37–44 (2005).
[26] Lin, S. and Chalupsky, H.: Unsupervised Link Discovery in Multi-relational Data via Rarity Analysis, *Proc. 3rd ICDM*, pp.171–178 (2003).
[27] Anyanwu, K. and Sheth, A.P.: The ρ operator: Discovering and ranking associations on the semantic web, *Proc. SIGMOD Record*, Vol.31, No.4, pp.42–47 (2002).
[28] Ward, J.H.: Hierarchical Grouping to optimize an objective function, *Journal of American Statistical Association*, Vol.58, pp.236–244 (1963).
[29] 坂野 鋭，山田敬嗣：怪奇!!次元の呪い：識別問題，パターン認識，データマイニングの初心者のために（前編），情報処理，Vol.43, No.5, pp.562–567 (2002).
[30] Duncan, J.W.: *Six Degrees: The Science of a Connected Age*, Heinemann, London (2003).

飯塚 京士 （正会員）

1993 年名古屋大学大学院理学研究科数学専攻博士前期課程修了．同年日本電信電話株式会社入社．NTT ソフトウェアイノベーションセンタ研究主任．セマンティック Web を用いた情報共有基盤およびクラウド技術の研究開発に従事．電子情報通信学会，人工知能学会各会員．2005 年度人工知能学会研究会優秀賞受賞．

村山 隆彦

1984 年東北大学工学部通信工学科卒業．1986 年同大学大学院工学研究科情報工学専攻修士課程修了．同年 NTT 入社．以来，知識処理，エージェント，Web サービス，セマンティック Web 等の研究開発に従事．2004～2010 年電気通信大学大学院情報システム学研究科客員准教授．電子情報通信学会，日本データベース学会各会員．



小林 透 （正会員）

1985 年東北大学工学部精密機械工学科卒業．1987 年同大学大学院工学研究科修士課程修了．同年 NTT 入社．以来，ソフトウェア生産技術，ユビキタスコンピューティング，情報セキュリティ，データマイニング等の研究開発に従事．現在，NTT サービスエボリューション研究所主幹研究員．電子情報通信学会（シニア会員），IEEE 各会員，博士（工学）．

赤埴 淳一

1983 年京都大学工学部数理工学科卒業．1985 年同大学大学院工学研究科数理工学専攻修士課程修了．1985 年 NTT 入社．スタンフォード大学計算機科学科ロボティックス研究所客員研究員（1989～1990 年），NTT コミュニケーション科学基礎研究所主幹研究員（1999 年），NTT 研究企画部門担当部長（2004 年），NTT ネットワークサービスシステム研究所主幹研究員（2008 年），NTT 情報流通プラットフォーム研究所主幹研究員（2010 年）を経て，2012 年 NTT アドバンステクノロジ株式会社担当部長．専門はセマンティックウェブ，エージェント指向プログラミング．