グラフデータに対するファセット探索のための頻出パターンを利用したオブジェクト抽出手法

駒水　孝裕† 　天笠　俊之††,††† 　北川　博之††

† 筑波大学大学院システム情報工学研究科 〒305-8573 茨城県つくば市天王台 1-1-1
†† 筑波大学システム情報系 〒305-8573 茨城県つくば市天王台 1-1-1
††† 宇宙航空研究開発機構 宇宙科学研究所 宇宙科学情報解析研究系
〒252-5210 神奈川県相模原市中央区由野台 3-1-1
E-mail: †taka-coma@kde.cs.tsukuba.ac.jp, ††{amagasa,kitagawa}@cs.tsukuba.ac.jp

あらまし　グラフは複雑なデータを表現可能なデータ構造である．一方で，ファセット探索は探索的な検索を行うための有用な手法として広く利用されている．グラフデータに対してファセット探索を行う際には，検索対象（ノードや部分グラフ）とプロパティ（特徴を表す属性）が予め定義されている必要がある．しかしながら，グラフデータ中の検索対象や検索対象のプロパティを予め指定することは容易ではない．これに対して本研究では，グラフデータに対するファセット探索を行うための検索対象の抽出方法について議論する．本研究ではグラフ中の頻出パターンに着目することで検索対象の抽出を行う枠組みを提案する．また，本稿では抽出した検索対象やファセットを効率的に格納する方法についても議論する．

キーワード　グラフ検索，ファセット探索，オブジェクト抽出，グラフデータ管理

Takahiro KOMAMIZU†, Toshiyuki AMAGASA††,†††, and Hiroyuki KITAGAWA††

† Graduate School of Systems and Information Engineering, University of Tsukuba
†† Faculty of Engineering, Information and Systems, University of Tsukuba
1-1-1 Tennodai, Tsukuba, Ibaraki 305-8573，Japan
††† Institute of Space and Astronautical Science, Japan Aerospace Exploration Agency
3-1-1 Yoshinodai, Chuuou, Sagamihara, Kanagawa 252-5210, Japan
E-mail: †taka-coma@kde.cs.tsukuba.ac.jp, ††{amagasa,kitagawa}@cs.tsukuba.ac.jp

1. はじめに

近年，大量かつ複雑な構造を持つデータが利用可能になり，このようなデータを記述するためにグラフがよく用いられる．グラフはノードとエッジによって構成されており，ノードは記述の対象を，エッジはノード同士の関係を表現する．記述対象の関係を記述することで単一の情報としては表現する事が難しい複雑なデータを表現する事ができる．グラフデータの例としては，ネットワーク（ソーシャルネットワークや Web，コンピュータネットワークなど），化学データ（化合物の構造など），地理データなどがある．

グラフデータベースはこのような多様なグラフデータを格納するためのデータベースである．例えば，facebook や Twitter などのソーシャルネットワークのデータや化合物を表すグラフのデータなどがある．前者ではグラフデータベースにはすべてのユーザアカウントから構成される一つのグラフが格納されるのに対し，後者では一つのグラフが化学式一つを表現するため，比較的小規模なグラフを大量に格納する．

これらのようなデータベースから効率的に必要な情報を検索することは，グラフデータを有効活用するうえで重要な課題である．これまでにグラフデータに対する検索については多くの研究がなされてきている [1]．例えば，比較的小規模なグラフが多数存在するようなデータベースでは，特定のパターンを持つグラフを検索する研究や入力したキーワードにマッチするグラフをランク付きで取得する研究などがある．しかしながら，これまでの手法では利用者が明示的にパターンをクエリ言語やプログラムを用いて記述しなければならない．また，キーワードによる検索も検索結果を更に絞り込む必要がある場合に更にキーワードを追加する必要があるが，絞り込みに効果的なキーワードを適切に入力できるとは限らない．そこで，利用者の検

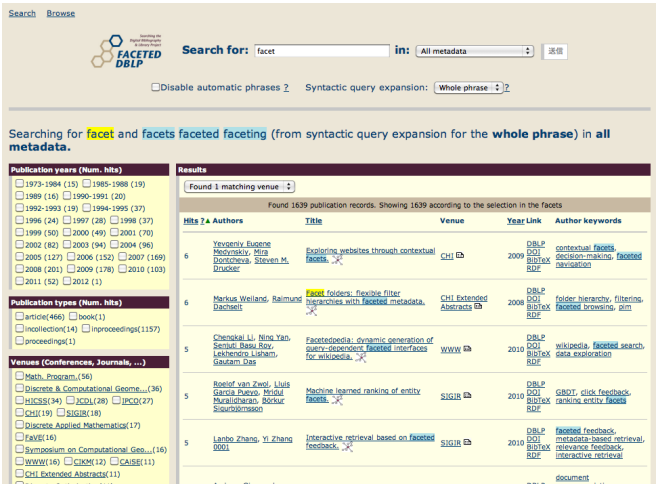

図 1　DBLP におけるファセット探索インターフェース(注5)

索を支援するための手法が必要である．

1.1　ファセット探索

ファセット探索 (*Faceted Navigation* または *Faceted Search*) はデータに対しインタラクティブなインターフェースを介して探索的に情報を検索する手法である [12]．ファセット探索は近年，図書館情報学や電子商取引 (*e-commerce*) などで注目を集めている技術である．利用されている例としては，DBLP(注1) (図 1) や IEEE Xplore(注2)，Amazon(注3)，ebay(注4) などがある．また，ファセット探索を他のデータの検索に対しても利用するための研究も盛んに行われている [7]〜[9], [15]．Yee ら [15] はファセット探索を画像検索に適用する手法を提案した．Yee らは画像に付加されたメタデータを利用してファセット探索を可能にした．Li ら [9] は Wikipedia に対する検索に対してファセット探索を適用する手法を提案した．Li らは Wikipedia に作成されているカテゴリを階層的なファセットと見てこれを利用したファセット探索を実現した．Koren ら [8] は Web 検索のインターフェースにファセット探索を導入する手法を提案した．Koren らは得られた検索結果からファセットを抽出し，それらを表示することで従来の検索で把握が困難だった Web 検索の結果を把握しやすくし，ファセット探索インターフェースを介して更なる検索を可能にした．また，我々は [7] で Twitter ユーザアカウント検索に対してファセット探索を可能にする手法を提案した．我々は Twitter 上の機能であるリストに着目し，ファセットを定義し，ファセット探索システムを構築した．

また，ファセット探索は現在の結果に対してファセットとその値のリストを返し，利用者はそのリストからファセットと値を選択することでさらに絞り込む．検索結果に応じてファセットと値のリストを生成するため，次にどんなファセットと値のペアを選択しても結果が空になることはない．この特性は極めて重要である．特にグラフのような複雑なデータの場合，利用者が更なる絞り込みのための条件を正確に記述することは，より深い専門知識や記憶力を必要とする．このような専門知識や記憶力は一般の利用者は持っていないため，ファセット探索による検索のサポートが有効である．

(注1)：http://www.informatik.uni-trier.de/~ley/db/
(注2)：http://ieeexplore.ieee.org/Xplore/dynhome.jsp
(注3)：http://www.amazon.com/
(注4)：http://www.ebay.com/
(注5)：http://dblp.l3s.de/?q=facet

1.2　本稿の位置づけ

利用者の検索を支援する手法として，我々はファセット探索をグラフデータの検索に適用する手法を提案する．本研究の全体像を図 2 に示した．本研究ではグラフデータ及びそれを格納するグラフデータが与えられた場合に，グラフデータベースに対してファセット探索を可能にする事を目的とする．この際の課題は次の様になる．(1) グラフデータベースから検索対象を抽出する．(2) 抽出した検索対象に対して検索を行うためにファセットを抽出する．(3) 検索対象及びファセットをファセット探索で利用可能な形で保管する．(4) 抽出した検索対象とファセットを用いてファセット探索を行うためのインターフェースを構築する．

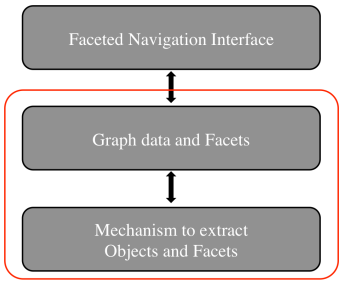

図 2　本研究の全体像（本稿では枠部分について議論する）

これらの課題に対して我々は，大規模なグラフデータに対してファセット探索を行うためのオブジェクトおよびファセットを抽出するフレームワークを提案する．ファセット探索では検索対象から抽出したファセットを利用して検索を行う．そのため，ファセット探索を行うためには検索対象を予め決める必要がある．[6], [10] で述べられる様にデータ内で頻出する構造は何らかのデータの単位を表している可能性が高い．そのために，本研究では頻出部分グラフに着目して検索対象の抽出を行う．

1.3　例示データ

本稿では，説明のために次の例を用いる．

[例 1](文献関連グラフ)　文献の関連を表すグラフの例を図 3 に示す．図中の各円はグラフ中のノードを表し，中の文字列はノードのラベルを表す．各四角はテキスト値を表し，点線矢印の原点ノードの値を表す．また，“→” は片方向エッジを表し，“↔” は双方向エッジを表す．例えば，図の一番左上の “author” と書かれた円は author ノードであり，その値が “John” ということを示している．さらに，この author は一つ右の author ノードから “supervise” ラベルが付加されたエッジと “coauthor” ラベルが付加されたエッジで差されていることで，右の author ノードと二種類の関係（supervise と coauthor）があることがわかる．

1.4　論 文 構 成

本稿は次のように構成される．まず，2. 節で本研究に関連する研究について紹介する．次に，3. 節で本研究の基本的な知識

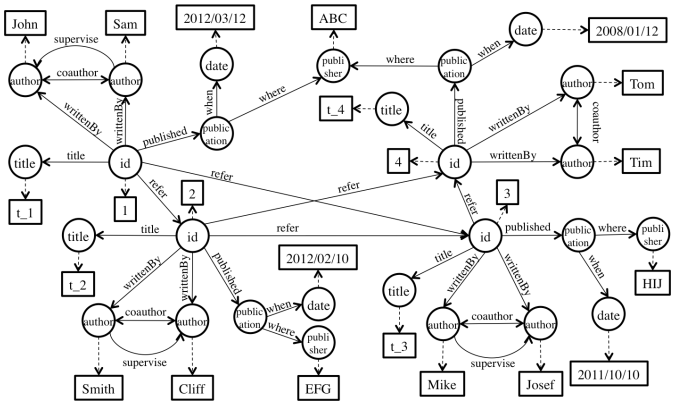

図 3　文献の関連グラフ

として，本研究で対象とするグラフデータや頻出部分グラフ，ファセット探索について説明する．続く 4. 節で本研究の提案手法について述べ， 5. 節で本手法の実装について述べる．また，6. 節で手法の有効性を確かめるための実験について述べる．最後に 7. 節で本稿のまとめと本研究の今後の課題について言及する．

2. 関 連 研 究

グラフデータに対するファセット探索に関連する研究が幾つか存在する．我々は XML データに対してファセット探索を適用する手法に関する研究を行なってきた [6]．加えて，グラフデータを記述するフォーマットの一つとして RDF (Resouce Description Framework)(注6) がある．RDF に対するファセット探索に関する研究として [4], [11] を紹介する．また，本研究で利用するグラフデータベースの種類として単一の大規模グラフに対するインタラクティブな検索手法に関する手法は著者らの知る限りでは存在しない．

2.1 XML データに対するファセット探索

先行研究 [6] で我々は XML データに対するファセット探索について研究を行った．XML データは木構造を持つ半構造データであるため，検索対象となる県債対象やその属性（ファセット）が一意に決まらない．そのため，XML データに対してファセット探索を適用するために検索対象及びファセットを決めるためのフレームワークを提案した．このフレームワークでは，XML データ中において親ノードに対して複数回出現するノードを検索対象とし，検索対象中でテキストを持つノードを検索対象の属性（ファセット）と定義した．この定義に基づき，検索対象およびファセットを抽出し，ファセット探索を実現するフレームワークを構築した．

2.2 RDF データに対するファセット探索

[4], [11] は RDF データに対するファセット探索を適用した．[4], [11] は両方とも条件にあったノードを検索することを目的としている．Oren ら [11] は，RDF の構造（主語，述語，目的語）に着目しファセットを述語とし，ファセットの値を目的語として定義した．さらに，この定義の上に検索を行うためのオペレーションを定義し，グラフを考慮したファセット探索を可能にした．Heim ら [4] は，RDF に対してグラフを意識したファセット探索インターフェース gFacet を構築した．gFacet は最初の段階で検索対象とするノードを指定する．その後は利用者が任意のタイミングで現在の検索結果のファセットのうち検索に利用したいファセットを表示することが可能なインターフェースとなっている．前述のとおり，これらの研究は単一のノードを検索することを主目的としている点で本研究とは異なる．

(注6)：http://www.w3.org/RDF/

3. 基 本 事 項

本節では，グラフデータに対するファセット探索に関する基本事項として，本研究で対象とするグラフデータとファセット探索について説明する．

3.1 グ ラ フ

本研究では，ラベル付き有向多重グラフを対象とする．ファセット探索において検索対象となる部分グラフを発見するため，グラフデータのインデクシングなどで用いられる頻出部分グラフを用いる．それぞれを 3.1.1 節と 3.1.2 節で説明する．

3.1.1 ラベル付き有向多重グラフ

ラベル付き有向多重グラフとは次の特徴を持つグラフである．(1) ノード及びエッジにラベルが付加されている．(2) エッジが方向性を持つ．(3) 二つのノード間に一つ以上のエッジが存在しても良い．ラベル付き有向多重グラフの定義を以下に示す．

[定義 1](ラベル付き有向多重グラフ)　ラベル付き有向多重グラフ $G = (V, E, L)$ はノード集合 V，エッジ集合 E，ラベル集合 L からなる．但し，マッピング関数 $\delta : V \cup E \rightarrow L$ はノード及びエッジへのラベルのマッピングを行う．また，各エッジ $e \in E$ は $e = (v_i, v_j)$ の様にノードのペアで与えられる．□

3.1.2 頻出部分グラフ

頻出部分グラフの定義を与えるために，まず部分グラフの定義（定義 2）を与える．あるグラフ G' の構成要素（ノード，エッジ）が一方のグラフ G の構成要素の部分集合となっており，ノード，ラベルのそれぞれについてラベルのマッピングが一致するとき，グラフ G' をグラフ G の部分グラフと呼ぶ．

[定義 2](部分グラフ)　グラフ $G' = (V', E', L')$ は次の条件を満たす時グラフデータ $G = (V, E, L)$ の部分グラフであると定義する．

（1） $V' \subseteqq V$
（2） $E' \subseteqq E$
（3） $\delta(v) = \delta'(v), \forall v \in V'$
（4） $\delta(e) = \delta'(e), \forall e \in E'$

但し，δ および δ' はそれぞれ G と G' のラベルのマッピング関数である．□

例えば，例 1 のグラフ（図 3）に対して，図 4(a)，図 4(b)，図 4(c) に示したグラフは部分グラフである．また，図 4(b) と図 4(c) のグラフは図 4(a) のグラフの部分グラフである．

次に頻出部分グラフの定義を与えるためにグラフ同型の定義（定義 3）を与える．ある二つのグラフにおいて，それぞれのノードにマップされたラベルが一致し，すべての対応するエッ

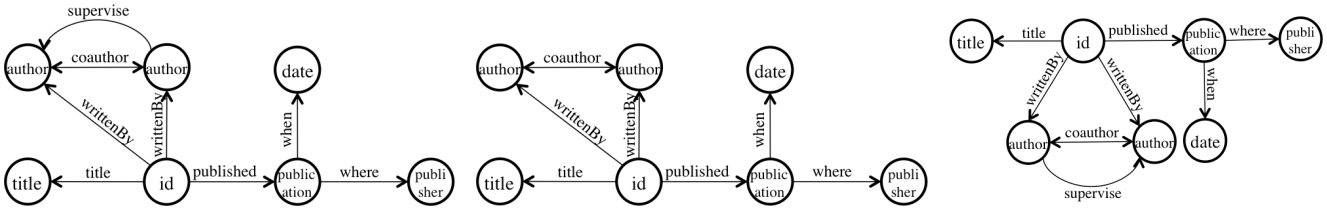

図 4　例 1 の部分グラフの例

ジについてラベルが一致するとき，これらのグラフは同型である．

[定義 3](グラフ同型) 与えられた二つのグラフ $G = (V, E, L), G' = (V', E', L')$ が次の条件を満たす時，グラフ G と G' はグラフ同型であると定義する．ここで，グラフ G におけるノード $v \in V$ と対応するグラフ G' におけるノードを $f(v) : V \to V'$ と記述する．

（1） $\delta(v) = \delta(f(v)), \forall v \in V$

（2） 各エッジ $e = (s, d) \in E$ について，$\delta(e) = \delta(e')$ となるような $e' = (f(s), f(d)) \in E'$ が存在する．

（3） 同様に，$e' = (s', d') \in E'$ について，$\delta(e') = \delta(e)$ となるような $e = (f^{-1}(s'), f^{-1}(d')) \in E$ が存在する．

但し，$f^{-1} : V' \to V$ は関数 f の逆関数である．□

例えば，図 4(a) のグラフと図 4(c) のグラフは同型であるが，一方で，図 4(a) のグラフと図 4(b) のグラフは同型ではない．

グラフ G におけるすべての部分グラフの集合のうち，同型なものを削除した集合を DG_G で表す．

[定義 4](重複なし部分グラフ集合) グラフ G における重複なし部分グラフ集合 DG_G を G の全ての部分グラフの集合から同型なものを取り除いた集合であると定義する．

最後に，頻出部分グラフは，与えられた回数（閾値）以上の頻度で同型の部分グラフがグラフデータ中に出現する部分グラフである．頻出部分グラフの定義を以下に示す．

[定義 5](頻出部分グラフ) 与えられたグラフ G と閾値 γ に対して，部分グラフ $G' \in DG_G$ が以下の条件を満たすとき，G' はグラフ G における頻出部分グラフである．

$$support_G(G') \geqq \gamma$$

但し，$support_G(G')$ をグラフ G における G' の出現頻度を表す．□

例えば，$\gamma = 4$ としたときに，例 1 のグラフ（図 3）G において図 4(b) のグラフ G' は $support_G(G') = 4 \geqq \gamma$ であるため，頻出部分グラフである．一方で，図 4(a) のグラフ G'' は $support_G(G'') = 3 < \gamma$ であるため，頻出部分グラフではない．

頻出部分グラフ集合のうち，グラフの大きさが極大の物を極大頻出部分グラフ集合とよび，以下のように定義する．

[定義 6](極大頻出部分グラフ集合) グラフ G における極大頻出部分グラフ集合 MFG_G を以下のように定義する．各部分グラフ $G' \in MFG_G$ が $support(G') \geqq \gamma$ を満たし，MFG_G 中に G' の同型な部分グラフのグラフが含まれない．

3.2 ファセット探索

ファセット探索は検索対象集合 O に対してファセット集合 F からファセットとその値を選択することで検索を行う．ファセット探索においてそれぞれのファセットは直交している．そのため，検索条件は現在までに選択したファセットとその値のペアで検索される検索対象集合の積集合となる．各ファセットと値の選択により得られた検索対象集合を $\sigma_i(O)$ とすると n 個の選択を行った検索結果は $\bigcap_{i=0}^{n} \sigma_i(O)$ となる．

4. 提 案 手 法

グラフデータに対してファセット探索を適用するための検索対象抽出手法を提案する．本手法の基本的な流れは次のようになる．まず，与えられたグラフデータからすべての極大頻出部分グラフ集合を抽出する（4.1 節）．得られた極大頻出部分グラフ集合から検索対象となる部分グラフを定義する（4.2 節）．次に，検索する際に用いる可能性のあるすべてのファセットを極大峻出部分グラフ集合を利用して定義する（4.3 節）．その後，実際のグラフから検索対象部分グラフ及びファセットをグラフデータから抽出する（4.4 節）．最後に，得られた検索対象集合とファセットの集合をデータベースに格納する（4.5 節）．

4.1 極大頻出部分グラフ集合の抽出

与えられたグラフデータベースから頻出部分グラフを抽出に関してこれまでに多くの研究が成されてきた．gSpan [14] や SpiderMine [16]，SUBDUE [5]，SEuS [3] などが代表的なものとして挙げられる．gSpan は頻出部分グラフの抽出手法としてはよく使われる手法で，小規模なグラフが大量に存在するようなグラフデータベースから頻出部分グラフの集合を抽出する手法である．また，SpriderMine は単一の大規模グラフデータを対象としているものの抽出する対象が k 個の最も大きい頻出部分グラフである．SUBDUE や SEuS は単一のグラフから頻出部分グラフ集合を抽出する手法であるが, [3] 中で SUBDUE が大規模グラフデータには向かないことが示されている．本稿では例 1 のような単一のグラフを例に手法を紹介するため，ここでは SEuS を用いて頻出部分グラフの抽出を行う．

SEuS はグラフの構造情報を用いて頻出部分グラフの抽出を行うアルゴリズムである．SEuS は次のフェイズからなる．(1) グラフデータの構造要約を作成する（要約フェイズ）．(2) 構造要約から頻出部分グラフの候補集合を生成する（候補生成フェイズ）．(3) 実際のデータに対し，頻出部分グラフの候補集合となった部分グラフの数え上げを行う（数え上げフェイズ）．

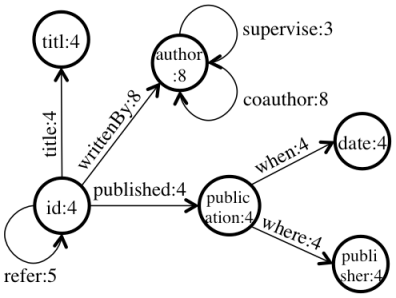

図 5　例 1 のグラフの構造要約

まず，(1) で与えられたグラフデータから構造要約を作成する，SEuS では構造要約にそれぞれのノードおよびエッジにデータ中の出現回数を付加する．例 1 のグラフデータの SEuS で用いられる構造要約を図 5 に示す．(2) では出現頻度の閾値により頻出部分グラフを決める．その際に，利用者とのインタラクションを介してパラメータを利用者に調整させることで精度向上を図っている．(3) 構造要約の情報だけによる部分グラフの出現頻度は正確でない可能性があるので，実際に部分グラフが頻出かどうかをチェックするために実データにおける数え上げを行う．なお，本研究で SEuS を用いる際には，候補生成フェイズにおける利用者とのインタラクションは行わず，比較的小さい閾値を設定し頻出部分グラフを抽出する．

頻出部分グラフはその要素数が爆発的に大きくなる可能性があるため，極大頻出部分グラフ集合を抽出する．これまでの過程で抽出した頻出グラフ集合から極大頻出部分グラフ集合を作成する．

4.2　検 索 対 象

ファセット探索を行うために，検索対象を決めなければファセットを決めることはできない．グラフデータベース D から得られた頻出部分グラフ集合を利用して検索対象となる候補を抽出した上で検索対象を決定する，検索対象を決定する方法として，例えば，システム開発者に頻出部分グラフ集合を提示し選択させる方法がある，システム開発者は極大頻出部分グラフ集合 MFG_D から検索対象としたい連結した部分グラフを選択する．

次に，各極大頻出部分グラフから，必要な部分グラフを選択させる．極大頻出部分グラフは頻出な部分グラフのうち極大な物を指しているため，実際は検索時にすべてのノードが必要であるとは限らない．そこで，極大頻出部分グラフ集合から検索対象となる部分グラフの集合 O を選択する．検索対象となる部分グラフ $o \in O$ は，極大頻出部分グラフ中に存在する連結な部分グラフと定義する（定義 7）．

[定義 7](検索対象)　ある極大頻出部分グラフ $mfg \in MFG_G$ から選択された検索対象となる部分グラフ $o \in O$ を以下の条件を満たす部分グラフと定義する．

（1）$o \subseteqq mfg$

（2）o は連結なグラフである．

□

4.3　ファセット

グラフデータ G において選択された検索対象となる部分グラフ集合 O に対してファセット探索を行うためのファセットを定義する．ある検索対象となる部分グラフ $o \in O$ からファセットとなりうるノードを抽出する．ファセットは値を持つ必要があり，また，値のセマンティクスを表す属性は値を直に所持するノードである可能性が高い．そのため，O 中のすべての検索対象となる部分グラフ o 中のノードでグラフデータ G 中で値を持つノードの集合をファセット集合とする（定義 8）．

[定義 8](ファセット集合)　与えられたグラフデータ $G = (V, E, L)$ と G における検索対象となる部分グラフの集合 O に対して，ファセットの集合 F を以下のように定義する．

$$F = \{v' \in V' \mid o = (V', E', L') \in O, v' \in V', \\ \delta(v) = \delta(v'), v.value \neq null\}$$

□

例えば，図 6 に示した検索対象となる部分グラフから得られるファセット集合は $F = \{title, id, author\}$ となる．

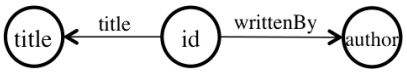

図 6　検索対象となる部分グラフの例

4.4　検索対象とファセットの抽出

本節ではファセット探索で用いる検索対象（部分グラフ）とファセットを選出および抽出する．ここまでは抽出された検索対象とファセットは極大頻出部分グラフ集合を元に自動的に抽出を行った．本節はこれまでに得られた極大頻出部分グラフ集合とファセットを利用し，グラフデータから検索対象の部分グラフおよびファセットの値を抽出する．

4.4.1　不要ファセットの排除

定義 8 に従い抽出されたファセットは必ずしも検索に向いているとは限らない．例えば，ID のような属性はファセットには向かない．あるいは，すべてのオブエジェクトが同じ値を持つファセットも検索に利用するのは難しい．このようなファセットを自動的に排除する．

まず．ID のように少数の検索対象を識別するようなファセットの排除について説明する．ファセット (f) における各値の出現頻度 $(freq(f.value))$ を計算する．この出現頻度の平均 (μ_f) を計算し，平均が以下の条件を満たす場合にファセット f を排除する．ただし，σ は予め決められた閾値である．

$$\mu_f < \sigma$$

一方のすべてが同じ値を持つようなファセットの排除方法について説明する．ファセット f における重複を除いた値の数 $(\mid distinct(f) \mid)$ が以下の条件を満たす場合にファセット f を排除する．ただし，ϕ は予め決められた閾値である．

$$\mid distinct(f) \mid < \phi$$

例えば，図 6 の検索対象 o_i におけるファセット集合

$F_{o_i} = \{title, id, author\}$ のうち id はファセットとしては利用しにくい情報である．なぜなら，id は文献情報の識別子であり o_i の検索対象インスタンスを一意に識別可能であるため，検索対象インスタンスも集合を絞り込むためのファセットには適さない．また一方で，ファセットの値がすべての検索対象インスタンスで同じ値である場合も同様にファセットに適さない．このような不要なファセットを排除する．

4.5 データベースへの格納

これまでにファセット探索に必要な検索対象及びファセットを選択した．本節では，検索対象インスタンス及びファセットのデータベースへの格納方法についてそれぞれ説明する．

4.5.1 検索対象インスタンスの格納方法

まず，検索対象インスタンスの格納方法について説明する．検索対象インスタンスはグラフデータベースの部分グラフであるため，検索の際にそれぞれの検索対象インスタンスにアクセスするためにはパターンマッチングを行わなければならない．しかし，大規模データに対するパターンマッチングはコストが大きい．そこで我々は，検索対象インスタンスを抽出しそれぞれを一つのデータとして扱いパターンマッチングを避けることでこの問題に対処する．最も単純な方法としては，それぞれの検索対象インスタンスに識別子を付加し，関係データベースに格納する方法である．その際のデータベーススキーマは以下のようになる．

object (object_id, object_instance)

4.5.2 ファセットの格納方法

次に，ファセットの格納方法について説明する．ファセット探索においてファセットを選択して検索を行う度に検索結果に対するファセットを表示する必要がある．また，ディスプレイに表示可能な情報が限られることから，それらのファセットの値は利用者が次に選択する可能性の高い上位 k 件が順に並べられることが望ましい．ファセットの順序付けはファセットの値の統計情報に基づいてお壊れることが多い．最も単純には，ファセットの値をそれを含む検索対象の数の多い順に並べる．このような処理に適したデータベースを用いることが望ましい．

本稿では，インデックスや集約演算をサポートする関係データベースを用いる．関係データベースに格納する際のデータベーススキーマは以下のようになる．

facet_name (value, object_id)

facet_name はファセットの名前をテーブル名にし，ファセット毎にテーブルを作成する．また，value はファセットの値を表し，object_id は検索対象インスタンスの識別子を表す．

5. システム実装

本手法を実装したシステムのアーキテクチャを図 7 に示す．本システムでは，FacetDB 及び ObjectDB には PostgreSQL(注7) を用いる．GraphDB としては HDFS (Hadoop Distributed File System)(注8) に JSON 形式で格納し Jaql [2] を用いてアクセスする方法を取る．前述のように，頻出部分グラフの抽出には SEuS アルゴリズムを用いる．

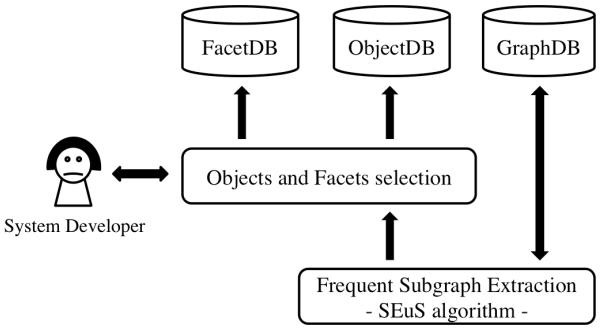

図 7 システムアーキテクチャ

システムの流れとしては，Frequent Subgraph Extraction モジュールが GraphDB からグラフデータを取得し，SEuS アルゴリズムを用いて頻出部分グラフを抽出し，Objects and Facets selection モジュールに受け渡す．Objects and Facets selection モジュールは受け取った頻出部分グラフをシステム開発者 (System Developer) に提示し，検索対象として利用する頻出部分グラフを選択させ，選択された頻出部分グラフを検索対象として ObjectDB に格納する．この際に各検索対象に一意に識別可能な識別子を付加する．次に，抽出された検索対象に対してファセットの候補を抽出し，システム管理者に提示し検索対象同様に選択させる．選択されたファセットを FacetDB に格納する．

6. 実データでの評価

我々は頻出部分グラフを抽出することで検索対象が抽出の可否を評価した．

利用したデータは Arnetminer(注9)で提供される DBLP の文献データと ACM citation の参照関係データを組み合わせて文献の参照ネットワークを作成した A Citation Network Dataset(注10) [13] を用いた．このデータのノード数は約 400 万個でエッジ数はおよそ 600 万個である．与えられるデータは以下のような形式で与えられる．

```
#*Spatial Data Structures.
#@Hanan Samet
#@John Smith
#year1995
#confModern Database Systems
#index25
#%165
#%221
#!An overview is presented of the... (detail omitted)
```

(注7)：http://www.postgresql.org/

(注8)：http://hadoop.apache.org/hdfs/

(注9)：http://arnetminer.org

(注10)：http://www.arnetminer.org/citation

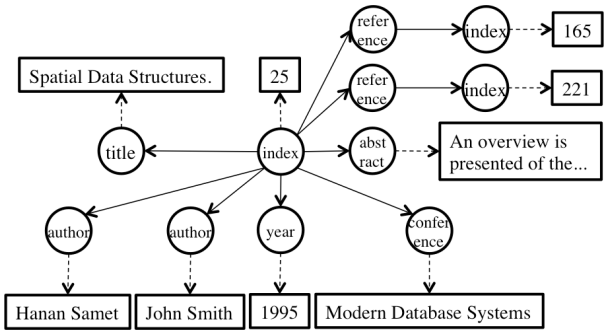

図 8　データ例の模式図

各行の接頭辞はそれぞれ次のようなデータを表す．`#*` は文献のタイトル (paperTitle) を表す．`#@` は著者 (Authors) を表し，複数出現することが可能である．`#year` は出版年 (Year) を表す．`#conf` は出版元 (publication venue) を表す．`#index` はデータセットないで一意な文献 ID (index id of this paper) を表す．`#%` は参照文献 ID (the id of references of this paper) を表し，複数出現できる．`#!` は文献の概要 (Abstract) を表す．このデータを模式化したものを図 8 に示した．

このデータから取得された検索対象の候補は図 9 の部分グラフである．このデータは文献の参照ネットワークであるため，文献が一つの単位として出現する回数が多いために文献に関連する属性の集まり（`index, title, author, year, counference, abstract, reference`）が唯一の検索対象の候補となった．

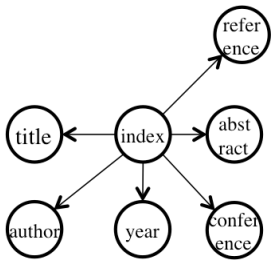

図 9　検索対象の候補

この検索対象に対し，ファセットの候補を抽出する．ファセットはテキストを持つノードと定義した（定義 8）．定義 8 に従いファセットを抽出すると次のようなファセットが得られる．index, title, author, year, conference, abstract, reference．このファセット集合のうちで index がファセットの自動選定によりファセットの候補から削除される．index を除いたこのリストをシステム開発者に提示し，選択を行う．今回は abstract と reference がファセットには向かないとして排除し，title, author, year, conference が図 9 の検索対象のファセットとなる．

このように今回用いたデータについては検索対象の抽出及びファセットの抽出を行うことが確認できた．

7. まとめと今後の課題

本稿ではグラフデータに対してファセット探索を行うための検索対象抽出手法を提案した．本手法は出現頻度に注目して部分グラフを検索対象にするための枠組みを構築した．また，それらをデータベースに格納剃る方法についても議論した．さらに，提案手法が利用である事が評価により示唆された．

今後の課題としては，本稿で構築したデータベースを用いてファセット探索を行うためのシステムを構築する．そのために，ファセット探索を行うメカニズムについて検討する．また，検索対象及びファセットを格納するデータベースに今回は関係データベースを用いたが他のデータベースと比較し，最適なデータの格納方法について検討する予定である．さらに，より複雑な構造を持つようなデータに対して本手法を適用し，利用の可能性を確認する．

謝辞　本研究の一部は科学研究費補助金 基盤 (A) (#21240005) による．ここに記して謝意を示す．

文　献

[1] Aggarwal, C., and Wang, H. *Managing and Mining Graph Data*. Springer, 2010.

[2] Beyer, K. S., Ercegovac, V., Gemulla, R., Balmin, A., Eltabakh, M. Y., Kanne, C.-C., Özcan, F., and Shekita, E. J. Jaql: A scripting language for large scale semistructured data analysis. *PVLDB 4*, 12 (2011), 1272–1283.

[3] Ghazizadeh, S., and Chawathe, S. S. SEuS: Structure Extraction Using Summaries. In *Proc. Discovery Science* (2002), S. Lange, K. Satoh, and C. H. Smith, Eds., vol. 2534 of *Lecture Notes in Computer Science*, Springer, pp. 71–85.

[4] Heim, P., Ertl, T., and Ziegler, J. Facet Graphs: Complex Semantic Querying Made Easy. In *Proc. ESWC* (2010), L. Aroyo, G. Antoniou, E. Hyvönen, A. ten Teije, H. Stuckenschmidt, L. Cabral, and T. Tudorache, Eds., vol. 6088 of *Lecture Notes in Computer Science*, Springer, pp. 288–302.

[5] Holder, L. B., Cook, D. J., and Djoko, S. Substucture discovery in the subdue system. In *Proc. KDD Workshop* (1994), pp. 169–180.

[6] Komamizu, T., Amagasa, T., and Kitagawa, H. A Framework of Faceted Navigation for XML Data. In *Proc. The 13th International Conference on Information Integration and Web-based Applications & Services (iiWAS)* (2011), pp. 28–35.

[7] Komamizu, T., Yamaguchi, Y., Amagasa, T., and Kitagawa, H. FACTUS: Faceted Twitter User Search Using Twitter Lists. In *Proc. WISE* (2011), A. Bouguettaya, M. Hauswirth, and L. Liu, Eds., Lecture Notes in Computer Science, Springer Berlin / Heidelberg, pp. 343–344.

[8] Koren, J., Zhang, Y., and Liu, X. Personalized Interactive Faceted Search. In *Proc. WWW* (2008), J. Huai, R. Chen, H.-W. Hon, Y. Liu, W.-Y. Ma, A. Tomkins, and X. Zhang, Eds., ACM, pp. 477–486.

[9] Li, C., Yan, N., Roy, S. B., Lisham, L., and Das, G. Facetedpedia: Dynamic Generation of Query-Dependent Faceted Interfaces for Wikipedia. In *Proc. WWW* (2010), M. Rappa, P. Jones, J. Freire, and S. Chakrabarti, Eds., ACM, pp. 651–660.

[10] Liu, Z., and Chen, Y. Identifying Meaningful Return Information for XML Keyword Search. In *Proc. SIGMOD Conference* (2007), C. Y. Chan, B. C. Ooi, and A. Zhou, Eds., ACM, pp. 329–340.

[11] Oren, E., Delbru, R., and Decker, S. Extending Faceted Navigation for RDF Data. In *Proc. International Semantic Web Conference* (2006), I. F. Cruz, S. Decker, D. Allemang, C. Preist, D. Schwabe, P. Mika, M. Uschold, and L. Aroyo, Eds., vol. 4273 of *Lecture Notes in Computer Science*, Springer, pp. 559–572.

[12] Sacco, G. M., and Tzitzikas, Y. *Dynamic Taxonomies and Faceted Search*. Springer, 2009.

[13] Tang, J., Yao, L., Zhang, D., and Zhang, J. A Combination Approach to Web User Profiling. *ACM Transactions on Knowledge Discovery from Data (TKDD) 5*, 1 (December 2010).

[14] Yan, X., and Han, J. gSpan: Graph-Based Substructure Pattern Mining. In *Proc. ICDM* (2002), IEEE Computer Society, pp. 721–724.

[15] Yee, K.-P., Swearingen, K., Li, K., and Hearst, M. A. Faceted Metadata for Image Search and Browsing. In *Proc. CHI* (2003), G. Cockton and P. Korhonen, Eds., ACM, pp. 401–408.

[16] Zhu, F., Qu, Q., Lo, D., Yan, X., Han, J., and Yu, P. S. Mining Top-K Large Structural Patterns in a Massive Network. *PVLDB 4*, 11 (2011), 807–818.