多次元的な Web 空間マイニングを行うデータベースシステムの実現: 一般化された制約条件への対応

齋藤 太陽 †　　大森 匡 †　　星 守 †

電気通信大学大学院情報システム学研究科 ‡

1. 研究の目的

近年，Web 上で仮想組織の活動が活発になってきており，Web 空間上の互いに興味を持ち合うページ集合 (コミュニティ) を抽出する Web コミュニティ分析の研究が盛んである．また，個人や状況に応じて情報をパーソナライズして調べることも重視されている [3]．筆者らは，多角的な分析視点から Web 空間マイニングを行うデータベースシステムを提案してきており，大学ドメイン内の分析を対象に，多次元的な制約組み合わせによる問い合わせに効率的に応答できることを示してきた [1]．本稿では，「大学の特定の分野ドメインを詳細に調べたい」，「指定キーワードに関連するコミュニティを詳細に調べたい」といった一般化された制約条件を用いた本システムの有効な利用法を示す．

2. 多次元的制約下のコア計算システムの概要

Web コミュニティ分析の分野では，Web ページをノード，リンクをエッジとした Web グラフにおいて完全 2 部グラフ（コア) を使ってコミュニティを求めることが多い．そこで我々は，調べたい視点を表す制約条件として，「情報系のドメインから見て重要なコミュニティを知りたい」「情報系と電気系のドメインの関係性から見て重要なコミュニティを知りたい」といった制約を考え，この制約を満たすコアからコミュニティを求めることにした．以下，システムの概要 [1][2] を述べる．

まず，準備として，Web グラフを図 1 左のような入辺型リンクレコード（以下，レコード) の集合で表したデータベースを用意し，指定した制約を満たすレコード集合からコア (図 1 右) を求めると考える．

次に，コア分析のために「FROM 型制約」と「TO 型制約」と呼ぶ 2 種類の制約条件を用意する．例として，分析対象とする Web 空間全体に領域別の階層構造を与える．例えば，電気通信大学 (uec.ac.jp) のトップドメイン (ALL) の下には，A:情報系のドメイン (ISJC)，B:電気系ドメイン (EE)，C:その他のドメイン (OTHER) がある．

FROM 型制約は，「どのドメインのページからリンクを張られているか」に着目した制約である．対象とする Web 空間を A, B, C の 3 つのサブドメインにわけて考えると，「FROM 型制約を A に設定する」とは，「ドメイン A のページを少なくとも 1 つ始点に持つレコードだけからコアを求める」ことを意味する．こうして求まるコアを，「FROM(A) 制約を満たすコア」と呼ぶ．図 2 の上部に，その例を示す．FROM(A) 制約を満たすコアは，

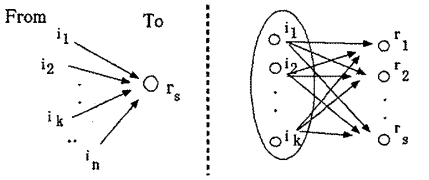

図 1　コアの計算方法

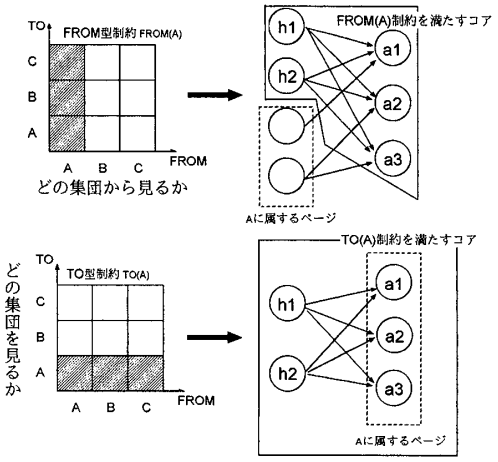

図 2　FROM 型制約と TO 型制約

ドメイン A から見たときに重要なコアを表す．

一方，TO 型制約は「どのドメインページにリンクを張っているか」に着目したものである．「TO 型制約を A に設定する」とは，「ドメイン A のページを終点に持つようなレコードだけからコアを求める」ことを意味する．こうして求まるコアを，「TO(A) 制約を満たすコア」と呼ぶ．このコアは，ドメイン A に属するページだけを終点として持つコアである (図 2 の下部)．TO(A) 制約を満たすコアは，Web 空間全体からドメイン A を見たときに重要なコアを表す．

与えられた FROM 型制約/TO 型制約を満たすコア集合 S を求めた後，そこからコミュニティ構造を表すグラフを作成して，重要なコミュニティを判定する．具体的には，まず，「共通する終点ページを 2 つ以上持つ」という条件が成立する (極大な) コア集合を S の中で求めて 1 ノード化し，これを 1 つのコミュニティと見なす．また，2 ノード間の有向辺を，当該ノードに含まれるページ間のリンクに応じて与える．こうして作った有向グラフ上で PageRank に準じたランク計算を行い，重要なノード (つまり，コミュニティ) を判定する．

以上の結果，例えば，FROM(A) 制約を満たすコア集合から，「ドメイン A から見たコミュニティのうち重要なものをランク順に求めよ」に答えることができる．

対象空間を A, B, C の 3 ドメインにわけて FROM 型制約と TO 型制約の 2 種類を制約として許す場合，コミュニティ計算として考えられる制約条件は，図 3 のようなデータキューブモデルで表すことができる．例えば，図 3 の Q3 は，「FROM(A or B) And TO(A or B)」という制約であり，これは，始点として A か B を持ち，終点が A か B になるレコードから求まるコア集合から作ったコミュニティ間の関係だけを考慮してランクづけするので，ドメイン A と B の間の関係性において重要なコミュニティを求めることになる．以上により，適当な分析用のスキーマを与えると，そのスキーマ上で FROM 型/TO 型の多次元制約問い合わせを行い，重要なコミュニティを求めることができる．これにより，例えば，「情報系ドメインから見るよりも空

Database System for Multi-Dimensional Web Mining

†T.Saito, T.Ohmori, M.Hoshi

‡The University of Electro-Communications

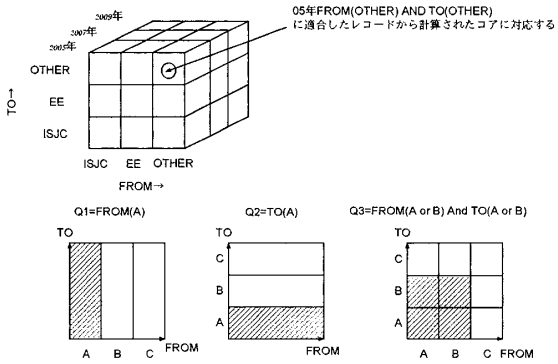

図 3　多次元制約下の問い合わせ例

間全体から見た方が順位の高い情報系コミュニティはどれか」などの分析ができる．

以上の提案システムは，対象リンクレコード集合の上で動作するデータベース処理システムであり，基本演算として，レコード集合からコア計算を行う実体化演算と，そこから特定の制約を満たすコア集合を取り出すフィルタリング演算，制約 1 を満たすコア集合と制約 2 を満たすものとから「制約 1 or 制約 2」を満たすコア集合を差分計算するマージ演算，などを持つ．そして，これらの組み合わせにより多次元的な問い合わせを処理する．

3. 一般的な制約述語を許したときの計算方法

多次元的なコミュニティ分析として利用価値を出すためには，上の例のように空間全体を 3 分割した単純なスキーマだけでなく，一般化された制約条件に基づいた多次元制約を扱う必要がある．ここで，一般化された制約条件とは「情報システム学領域 (IS) のみ」，「OTHER のうち機械関連でないもの」や，「指定キーワード K を満たすページから前方 N ホップ以内にある」というような制約条件のことである．正確に言うと，ページに関するブール述語 $p_i(i = 5, 6, 7, \ldots)$ を与えた時に，p_i を満たすページを始点 (または終点) に持つリンクレコードを使って，図 2 の場合と同様に FROM(p_i) 制約 (または TO(p_i) 制約) を満たすコアを定義し，これらの FROM/TO 型制約による多次元制約下のコア計算を行いたい．

これら一般的な制約条件は，2 節の 3 領域分割のときのように対象空間全体を被覆しないし，直和分割にもなっていない．また，これらの一般的制約条件を事前に予想することはできないから，利用者が実行時に与えることが想定される．さらに，コア計算では，通常，計算に用いる内部リンク数 (同一サイト内のリンク) を厳しく制限し，求めたいコアの始点数/ 終点数にも制約をつけないと実質的ではない．本システムは，b 本以上の内部リンクのみしか持たないリンクレコードを計算対象から外し，コアの終点数 s 以上を全て求めることにしている．このとき，$p_5, ..., p_8$ から成る多次元制約に応じた問い合わせ実行方法を決める必要がある．uec.ac.jp の場合，空間全体 (ALL) のコアは $b = 8, s = 4$ で計算して保持し，その下の 3 分割階層については，$b = 12, s = 4$ で FROM(X) (X=ISJC, EE, OTHER) の制約下でコア集合を計算して保持することができるが，これらの b よりも大きな値では計算できない．一方，$p_5, ..., p_8$ の一般化制約条件では，b, s といったコアの細かさを指定するパラメータをより厳しくした場合を想定する．例えば，b=16 や 20 では空間全体 (ALL) や FROM(OTHER) のコアは計算コストが高過ぎて予め用意できないか用意することが現実的でないが，一般化制約条件なら可能となる場合である．例えば，実際に uec.ac.jp の中で「指定キーワード K を満たすページから前方 N ホップ以内にある」という述語なら，$b = 16 - 20$ でも効率良くコア計算できる．そこで，一般化制約述語 p_i については，必要になったときに FROM(p_i) や TO(p_i) を計算して維持し，実行中の変更 (削除や追加) も許すことにする．

このとき，一般制約条件について，FROM(p_5) と FROM(p_6) が既に保持されていれば「FROM (p_5 or p_6) And TO(p_5 or p_6)」の問い合わせは，直接再計算が不要である．つまり，FROM(p_5) から「FROM(p_5) and TO(p_5 or p_6)」を満たすコア集合を取り出して (フィルタリング演算)，同様に FROM(p_6) からフィルタリング演算を行った結果との間で，マージ演算を行うことになる．(いずれの演算も，コア集合に対する変形や差分計算処理を伴う操作であり，自明な処理ではない [1])．

4. 問い合わせの事例

一般化制約条件の例として，p_5, p_6 を，それぞれ，IS(情報システム学部門) のみ，J(計算機科学部門) のみという述語に設定し，コアの詳細度を b=16, s=4 に固定した場合に分かるコミュニティの内容を調べた．b=8 と b=16 の FROM(IS or J) And TO(IS or J) の結果に基づき，上位 20 位を Jaccard 類似度 (4% 以上) で比較した．b=8 と b=16 の順位を比較した対応表を図 4 に示す．

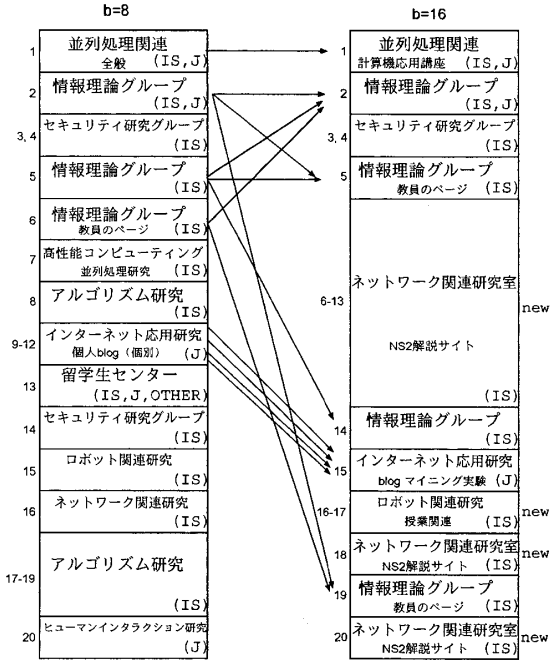

図 4　b=8 と b=16 の Jaccard 類似度 (4%) による比較

1 位のサイトについて，b=8 の場合，並列関連ページが集まったコアコミュニティノードであったが，b=16 で見た時，その中のある特定の研究室サイトが目立って現れていることに気づくことができた．また，b=16 で詳細度をあげてみた場合に 15 位に出てきたコアコミュニティノードは blog マイニングに使われる実験用の blog サイトであった．これは，b=8 の 9 位から 12 位に出ていた個人の個別 blog と対応していることが分かった．このように，特定分野についてパラメタ b を変えて内部リンクをより詳しく考慮することで，自明でないコミュニティを見つけられる．この他の述語，例えば，「指定キーワードに照合するページの周辺のコミュニティを求めよ」，などの多次元制約を使った場合の効果や問い合わせ処理方法の詳細は文献 [2] で述べる予定である．

参考文献

1. 栗原, 大森, 星, “Web 構造分析を目的とした多次元データマイニング構造の効率化,” DEWS2008, D1-6, 2008.

2. 齋藤, 大森, 星, “多次元的な Web 空間マイニングを行うデータベースシステムの実現,” DEIM 2010 投稿中.

3. S.Raghavan, H.Garcia-Molina, “Complex Queries over Web Repositories,” VLDB 2003, pp.33-44, 2003.