D-017

グラフデータベースにおけるキーワード検索の結果分類方式
Classifying Top-K Answers of a Keyword Search Algorithm on Graph Databases

金　銀実†　　大森　匡†
Yinshi　Jin　　Tadashi　Ohmori

1. はじめに

近年，関係データベースのような構造化データや XML などの半構造データの集合をグラフデータで表してキーワード検索する研究が注目されている[1][2]．例えば，関係データベースのタプルや XML のタグを頂点，関係データベースの外部キー参照や XML の階層関係を辺と見なせば，全てのデータを 1 つのデータグラフに統合して表せる．データグラフでのキーワード検索とは，与えられたキーワードを含む適切な部分グラフを答えとして求め，適当な基準でランクづけし，上位 k 個(top-k)の答えを返す検索である．一方，この種の研究では，スキーマを知らないユーザがキーワード入力だけで検索するため，答えの構造が異なる部分グラフが多数返答され，答えの理解自体がユーザには難しい．そこで本稿では，関係データベースをグラフ化して star-tree を答えとするキーワード検索を取り上げ，この問題の解決を目指す．

2. データグラフ

データグラフは$G=(V,E,w)$で表わせる．Vは頂点(関係データベースのタプル)の集合，Eは頂点と頂点をつなぐ無向辺(関係データベースの外部キー参照)の集合である．また，wは各エッジ$e \in E$に対する，正の重み$w(e)$を表す重み関数であり，重みが小さいほど頂点間の関係性が強い．

3. 答えとしての極小連結木

データグラフに対するキーワード検索は，各キーワードを少なくとも一回含む極小連結木(MCT)を答えとし，コストの小さい順に上位 k 個を検索する方式がある．

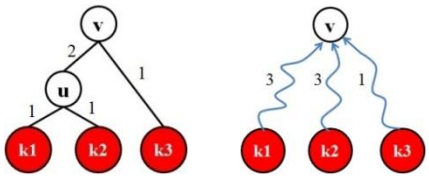
3.1 極小連結木(MCT)とは

極小連結木とは，全部のキーワードを含む連結木で，一つでも頂点を取り除いた真部分木は全部のキーワードを満たせないか連結にならないことである．しかし，一般的なグラフで MCT を検索することは，スキーマが設計されている RDB 或いは XML 上での検索に比べ,計算困難な問題である．これは NP 完全である GST(Group Steiner Tree)問題を含む．本稿では d-star tree で MCT のコストを近似する．

3.2 d-Star Tree Problem

d-star tree による近似コスト：キーワード集合$Q=\{k_1,\ldots,k_l\}$が与えられたとき，データグラフ$G=(V,E)$の任意の頂点$v \in V$を根とする木 T のコストを，頂点vから各キーワード$k_i(i=1,2,\ldots,l)$を満たす頂点までの最短距離の和　$s(T)=\sum_{i=1}^{l} distance(v,k_i)$ で与える．

図 1 に示すように steiner tree によるコスト 5(図 1(a))の MCT は，式(1)の d-star tree のコスト計算方式によるとコストは 7(図 1(b))である．計算アルゴリズムは文献[1]に従い，木の高さ制限 d は行っていない．



(a) steiner tree(コスト:5)(b) d-star tree(コスト:7)

図 1：MCT のコスト

3.3 問題点

検索した結果のうち，どの検索結果がユーザにとって有用なのか，その検索結果をユーザにどのように説明するのかが本稿の問題である．つまり，ユーザが知りたい情報が分らない場合，検索した情報をどのように「表示」，もしくは「通知」したら,ユーザ側から見てより分りやすい,判断しやすいのかを考えることが解決すべき課題である．以下，上位 k 個の解をより分かりやすく説明できるように，適切な解釈モデルを考え，それに基づいた解のグループ分けを行う方法を述べる．

4. 検索結果のグループ分け

例として，図2の論文データベースは，図3のようにConference，Session，Paper，Authorの4つのエンティティで構成されたグラフデータベースによって表せる．(図3の無向辺は全てw=1の双方向辺とした.)

問い合わせQ={VLDB(k0)，keyword(k1)，Sanjay(k2)}が与えられたとき，ノードV00，V04，P2，P4，A1がそれぞれのキーワードを含むノードである．Qに対するtop-6の解を図4に示す．これらの情報をより分かりやすく説明するため，キーワードを含む葉ノードから根ノードまでの距離を使った解釈モデルによるグループ分けを考える．

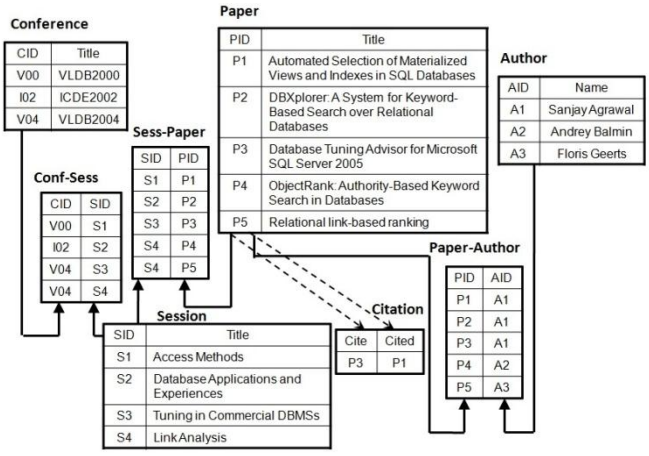


図 2：例題論文データベース

†電気通信大学大学院情報システム学研究科
Graduate School of Information Systems, The University of Electro-Communications

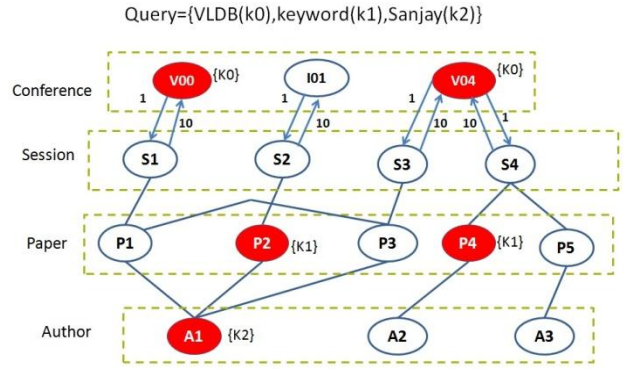


図 3：例題グラフデータベース

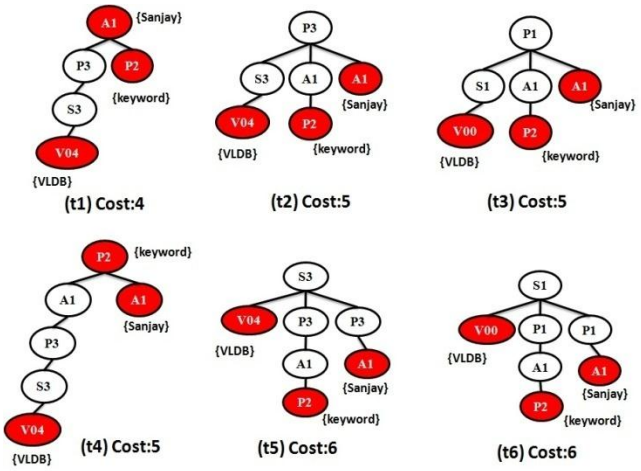


図 4： Q={VLDB(k0)，keyword(k1)，Sanjay(k2)}の top-6 解

（分類法１）答えとなる木Tにおいて，キーワードを持つ葉ノードから根ノードvへの距離を小さい順にdx，dy，dzとおき，該当するキーワードをKx，Ky，Kzとおいて，Tをグループ<dx，dy，dz>に分類する方法を考える．これは，解釈モデルとしては，「キーワードKxに近いエンティティ（ｖのこと）があり，他に(近い順に)Ky，Kzにも関連する」という意味でTを分類したことになる．例えば，3キーワードの検索で答えとなる木Tについて d2 < d0 < d1 なら，Tを<d2，d0，d1>のグループに分類する．さらに，同じ1グループに属す木の集合は，3.2節の木のコスト定義に基づき，コストの小さい順にソートして管理する．

この分類法を使うと，図4のtop-6の解は図5のように4グループに分けられる．図中，グループ１ <d2,d1,d0>は，「Sanjayに関する情報でkeywordに近く，VLDBにも関連した答え」を意味する答えのクラスタである．このグループ１に属す木t1は下記の内容である：

t1：「著者A1{Sanjay}は{keyword}を含む論文P2を書いていて，その著者{Sanjay}は別の論文P3を書き，その論文はS3のSessionに属し，そのSessionはVLDB2004に属す」．

同様に，グループ4 <d0,d2,d1>は，「VLDBの情報でSanjayに近く，keywordに関連するもの」を意味し，所属するt5は，「VLDB2004のSession S3に論文P3があり，その論文P3は著者Sanjayによって書かれ，その著者はkeywordを含む他の論文P2を書いている」という情報である．答えt6

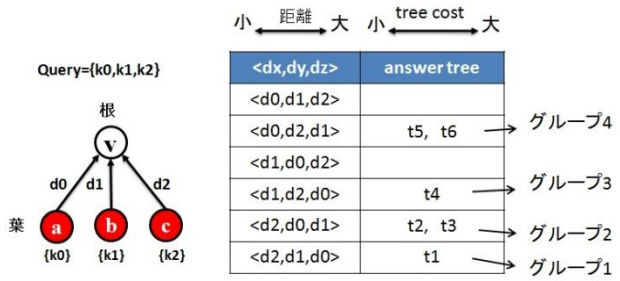


図 5：答えになる木のグループ分け

もt5と同じ構造(エンティティの木構造)を持つ解である．この例からも分かるように，グループ2のt2とt3（または，グループ4のt5とt6）は，図2への1つのSQL文から生成される答である．すなわち，分類法1では，1つのcandidate network([2])から生成される答えは必ず同一グループに所属する．そのため，同一のSQL文の結果に相当する答えが上位K解に多数ある場合には有効である．

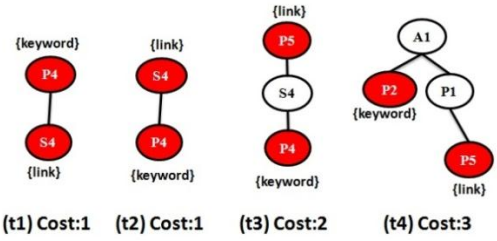


図 6： Q={ keyword(k0)，link(k1)}の top-4 解

（分類法2）同一グループ<dx,dy,dz>に構造の異なる木が混在する場合も答えが理解しにくい原因の一つである．このとき，分類法１ではまだ足りない．そこで，<dx,dy,dz>に，木Tの根ノードｖが属すentity情報を追加し，<dx,dy,dz>に属す木をさらにその根ノードの<entity>で細分類することにする（分類法2）．例えば，図3のp1-p5間にcitation辺（重さ１）を追加してQ={keyword(k0),link(k1)}を問い合わせとしたときの上位4解を図６に示す．グループ<d0,d1>には木t1とt4が入るが，keyword(k0)に近い根ノード情報がエンティティpaperかauthorかで異なるグループを作ることになる．この方法では，グループ<d0,d1><author>は「k0に近いauthorでk1に関連する答え」という解釈モデルとなり，分類法１よりも具体的な情報をグループごとにユーザに提示できる．従って，スキーマ情報に基づいて多数のcandidate networkからの解が上位K解に入るときにユーザがグループ選択を行うには有効である．

5. まとめ

本稿では，グラフデータベースの top-k キーワード検索技法で答えとなる部分グラフをできるだけ共通の解釈モデルになるよう分類する方法を述べた．上位 K 解の分類解釈提示は高品質な解の探索に必要な機能の一つであり，より複雑なスキーマでの試行結果は講演時に報告したい．

[参考文献]

[1] M. Zhong, M.Liu, “Efficient Keyword Proximity Search using a frontier-reduce Strategy based on *d*-Distance Graph Index,” IDEAS'09, pp.206-216 (2009).

[2] Z.Zeng, et al., “iSearch: An Interpretation based Framework for Keyword Search in Relational Databases,” KEYS 2012, ACM (2012)