様　式　Ｃ－１９、Ｆ－１９、Ｚ－１９　（共通）

科学研究費助成事業　　研究成果報告書

平成 ２７ 年　 ６ 月 ２３ 日現在

機関番号：６２６１５
研究種目：挑戦的萌芽研究
研究期間：2012～2014
課題番号：２４６５００４２

研究課題名（和文）大規模無順序木データベースのトップK検索アルゴリズムの研究

研究課題名（英文）A Study on Top-K algorithm for Large Unordered Tree Databases

研究代表者
高須　淳宏（TAKASU, Atsuhiro）

国立情報学研究所・コンテンツ科学研究系・教授

研究者番号：９０２１６６４８

交付決定額（研究期間全体）：（直接経費）　　3,000,000円

研究成果の概要（和文）：本研究は、ＸＭＬ文書や数式など、さまざまなデータを計算機で扱う場合に用いられる木構造データの効率的なマッチングおよび検索を実現するための基盤技術の研究開発を行った。特に、コストの高い計算を要する無順序木の処理アルゴリズムの研究を行った。研究成果は効率の良いマッチングアルゴリズムの開発と大規模木構造データベースの索引に分けられる。まず、木の幅が狭い場合に効率よく処理可能な無順序木のマッチングアルゴリズムを提案し、中規模データに対して実用的な時間で計算可能なプログラムを作成した。また、木構造の類似度に重点をおいた木構造データベースのメトリック空間索引を開発した。

研究成果の概要（英文）：Trees are used for representing and processing various data such as XML documents and mathematical formulas. We studied efficient tree matching and retrieval algorithms. This study focuses on the algorithms for unordered trees that generally require high computation cost. We first proposed an unordered tree matching algorithm that is especially effective for narrow trees and developed a program that can calculate the similarity of mid sized trees within reasonable processing time. For processing large tree databases, we developed efficient indices that can detect candidate trees from the database. For the case that tree structure is important for retrieval, we made a metric space-based index that converts each tree to a feature vector then makes a metric space for the vectors. Then, we apply a pivot-based indexing technique.

研究分野：情報工学

キーワード：木構造データ検索　トップK検索　インデキシング

１．研究開始当初の背景

木構造は、構造を持ったデータを表す基本的なデータ構造として、ＸＭＬ文書、画像データの解析結果、数式、遺伝情報学の糖鎖データなど、さまざまなデータを計算機で扱う場合に用いられる。そのため、木構造データを管理するデータベースシステム、木構造データのマッチングや検索、木構造データのマイニングなど、木構造データに関する様々な問題についての研究が行われている。木構造データの検索やマイニングの基本機能である木構造データのマッチングはコストの高い計算を必要とするため、マッチングの高速化と大規模木構造データからの類似木の検索の高速化は木構造データの処理において必須の課題となる。

木構造データのマッチングは、Combinatorial Pattern Matching の分野でこれまでに精力的に研究されてきた問題で編集距離等の木の距離・類似度を定義した上で効率の良いマッチングアルゴリズムが開発されてきた。木のマッチングにおいて、木の各内部頂点の子頂点の順序が固定されている順序木の編集距離は、頂点数を n として $O(n^3)$で解けることが知られている。一方、内部頂点の子頂点の順序が固定されていない無順序木の場合、編集距離の計算は MAX SNP 困難な問題であることが知られている。無順序木になると編集操作の可能性が指数的に増え、計算が難しくなる。この計算複雑性が、大規模無順序木データベースに対する処理の障壁となってきた。そのため、無順序木としてデータを扱う必要がある場合は、対象データの特性にあわせて、効率の良いマッチングアルゴリズムを用いることが重要になる。

木構造データの距離・類似度を測るためによく用いられる編集距離では、挿入、削除、置換の３種類の操作を用いて木の変換を行う。２つの木構造データの編集距離は、上記３種類の操作を適用して、一方のデータから他方のデータに変換する場合に必要となる最小の操作数と定義される。通常の編集距離では、挿入・削除・置換の操作には同一の重みが用いられるが、実際のデータや応用を考えた場合に、各種操作の重みを問題に併せて調整することが望ましいことがある。問題に適した重みを獲得するため、類似木から構成される訓練データを用いて各操作の重みを学習する方法がこれまでに研究されてきた。

データベースに含まれる木構造データの数が多い場合は、２つの木構造データの距離を効率良く計算するアルゴリズムの研究開発に加えて、大量のデータの中から類似する木構造データを効率良く絞り込む技術も重要になる。このために木構造データベースのインデキシングの研究が行われてきた。例えば、ザルツブルク大学の Augsten 等は、pq-gram と呼ばれる部分木を用いたインデキシング法を提案した。また木構造データを文字列データに変換した上で、文字列データの索引技術を用いた検索法も提案されてきた。木構造データの索引に関する研究は、順序木を対象としたものが多く、無順序木を扱う場合は、新たなインデキシング技術が必要になる。

２．研究の目的

本研究は、大規模木構造データベースを活用するためのマッチング・検索技術を開発することを目的としている。主に無順序木を対象とし、多様な問題に適した木構造データの距離・類似度の計算法と効率の良いアルゴリズムの開発を行う。また、大規模木構造データベースにも適用可能な技術とするため、木構造データのインデキシング法の研究開発を行う。これにより、自然言語処理を通して得られる、より精緻なテキストの特徴を用いた情報の検索、XML などの半構造データベースのスキーマ統合、多様な形式で記述される数式の構造情報を用いたマッチング・検索、遺伝情報の検索など、木構造を考慮したマッチングが有効な問題に対する共通の検索基盤技術を構築することを目的とした。

３．研究の方法

本研究では、これまでに研究者代表者等が順序木を対象として行ってきた木構造データのマッチング・検索技術を出発点とし、その技術を改良するとともに、無順序木に展開することによって、以下の２つの課題を中心に大規模木構造データ検索基盤技術の研究開発を行った。

(1)木構造データ間の距離計算法と効率的な計算アルゴリズムの研究開発

編集距離に基づいた木構造データの距離の計算法について検討を行った。無順序木の類似度の計算アルゴリズムについての理論的な分析、対象データや検索タスクに適した編集コストの学習法、半構造テキストデータの統合に適した木構造データの類似度の計算法と効率の良いアルゴリズムの研究開発に取り組んだ。

(2)大規模木構造データベースのインデキシング

木構造データベースの検索において問い合わせ木に類似した木を効率良く絞り込むためのインデキシング法についての研究を行った。木構造データの計算効率の良い特徴ベクトル化および近似距離計算のためのピボット選択問題に取り組んだ。また、半構造テキストデータに適したフィルタリング法の研究を行った。

４．研究成果

(1) 木構造データ間の距離計算法と効率的な計算アルゴリズムの研究開発

木構造データ間の距離の計算については、以下の３つの課題に取り組んだ。①無順序木のマッチングの計算量についての理論的分析を行い、木の最大次数が小さい場合は効率良く計算できることを示した。②無順序木の実用的な計算法を提案し、100 程度のノード数の木に対して実用的な時間で計算可能なプログラムを作成した。③順序木のアライメントのための編集コストを訓練データより求める方法の改良を行い、テキストデータを葉に持つデータに対する編集コストの学習を可能にした。

①無順序木マッチングアルゴリズムの理論分析

無順序木の編集距離は、NP 困難な問題であることが知られている。本研究では、対象とする木構造データの特性とマッチングアルゴリズムの関係について検討を進めた。マッチングアルゴリズムの評価を行う場合、木構造データのノードの数を n とし、n に対して必要となる計算量を評価するのが一般的である。しかし、木構造データはノード数が同じでも、深さの深い木や幅の広い木など、さまざまな木が存在する。本研究では、木構造データの幅が狭い場合のマッチングの計算量について、理論的な分析を行った。

本研究では、無順序木の最大共通部分木を求める問題をとりあげた。図１は、２つの木 T_1,T_2 に対する最大共通部分木(LCST)の例を示している。２つの木のノードを破線で示されるマッピングで対応づけると図下部に示される共通部分木が得られる。この例では、この共通部分木が最大の部分木となっている。

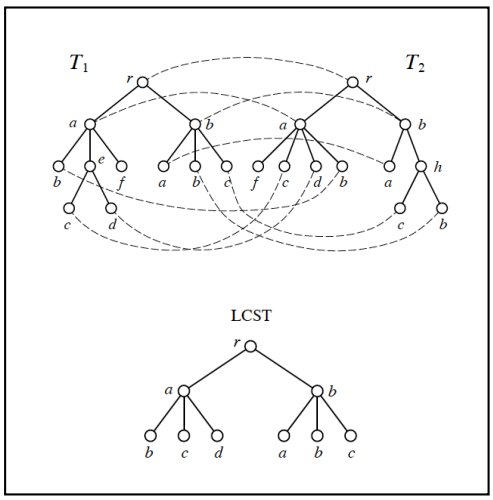

図 1　最大共通部分木

木の最大次数が D の場合、動的計画法を用いることによって、計算量 $O(n^{2D})$で計算可能なことを示した [1]。この計算量は、最大次数 D が小さい場合には、現実的な時間で計算可能なことを示している。

② 無順序木の実用的マッチングアルゴリズム

実用的な観点からは、ある程度のサイズの無順序木のマッチングを現実的な時間内で求めるソフトウェアが必要になる。本研究では、最大クリークを計算するライブラリを用いた無順序木の編集距離を計算するプログラムを開発した。

以下の図に最大クリークを用いた編集距離の計算法の例を示す。２つの木(T_1,T_2)に対して、マッピング可能なノードの対をノードとするグラフ G を作成する G におけるクリークは、２つの木のマッピングに対応する。例えば、図２のクリーク{(a,q),(b,r),(d,p)}は、図の左下に示されるマッピングと対応する。マッピングと編集操作を対応づけることが可能であるため、編集距離を計算する問題は、グラフのクリークを検出する問題に変換することが可能になる。最小編集操作に対応するようにグラフ G を重み付けすることによって、編集距離の計算が可能になる。

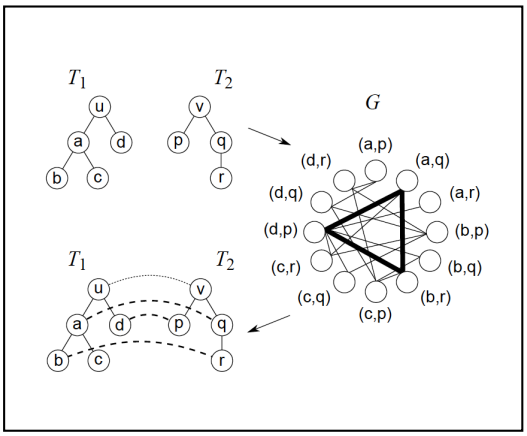

図 2　最大クリークと木編集距離

最大クリークは、汎用性の高い問題であり、これまでに様々な効率の良い solver が開発されてきた。本研究では、電気通信大学の中村等が開発した solver を用いてプログラムを開発した。京都大学が所有する糖鎖データを用いて性能評価を行ったところ、サイズが 80～85 の木に対しても 10 秒程度で無順序木の編集距離の計算が可能であることがわかった。表１は、木の大きさと計算時間（秒）の関係を示している。

total number of nodes	CliqueEdit	DpCliqueEdit
30 ~ 34	0.00434	0.00556
35 ~ 39	0.00499	0.01180
40 ~ 44	0.01520	0.02080
45 ~ 49	0.05080	0.03830
50 ~ 54	0.47300	0.13000
55 ~ 59	2.16000	0.12900
60 ~ 64	3.02000	0.19900
65 ~ 69	15.30000	0.40900
70 ~ 74	4.38000	0.84600
75 ~ 79	2.61000	1.12000
80 ~ 84	7.93000	2.84000
85 ~ 89	232.00000	64.70000

表 1　木編集距離の計算時間

③編集コストの学習

データの特性や処理目的に適した類似度・距離を得るための編集コストの学習法について研究を進めた。本研究では、研究代表者等が以前に行った類似木生成モデルの研究を発展させ、２つの木のアライメント木を生成する確率モデルを用いた学習法を考案した。確率モデルとしては、確率文法や木オートマトンなどが使用できるが、本研究では研究代表者等が以前に提案した隠れマルコフモデル(HMM)を拡張したモデルを用いた。図３は、木のペアを生成する確率モデルを表している。図３の左パネルがアライメント木の子孫を生成する隠れマルコフモデルを表している。中パネルはこのモデルを用いて子ノードを生成した例を示している。一方、右パネルは葉ノードを生成するモデルを表している。図中のτ、π、οは、モデルのパラメタを表している。

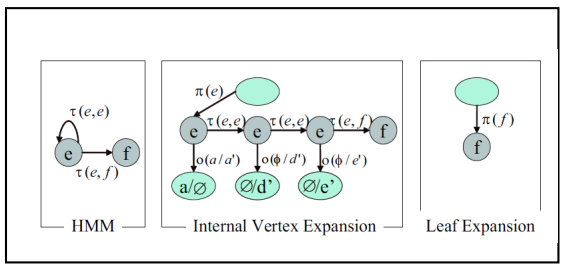

図 3 アライメント木生成 HMM

モデルのパラメタは、類似木の集合として与えられる訓練データを用いて学習する。以前の研究では、最尤推定およびベイズ推定を行うための EM アルゴリズムを開発した。本研究では、訓練データがアライメント済みの類似木を含んでいる場合の教師付きデータに対する学習アルゴリズムを開発した。

数式１は、学習アルゴリズムで最大化する目的関数を示している。式中の $P(A|T,S,\Theta)$ は、パラメタΘのもとで、訓練データに含まれる類似木のペア T,S に対するアライメント A の確率を示している。また、σ^2は、パラメタ推定の正則化のパラメタを示している。

$$
\begin{aligned}
& L(\Theta) \\
&= \log\left(\prod_{(T,S,A)\in D} \mathrm{P}(A \mid T,S,\theta)\right) - \sum_{\theta\in\Theta}\frac{\theta^2}{\sigma^2} \\
&= \sum_{(T,S,A)\in D} [\log(\mathrm{P}(T,S,A)) - \log(\mathrm{P}(T,S))] \\
&\quad - \sum_{\theta\in\Theta}\frac{\theta^2}{\sigma^2} \ .
\end{aligned}
$$

数式 1　パラメタ推定の目的関数

(2)大規模木構造データベースのインデキシング

木の類似度を計算する場合、木の構造情報を主要な特徴として類似度を計算する場合と、木のノードに付与されたテキストを主要な特徴として類似度を計算する場合が考えられる。

本研究では、木の構造情報に着目したインデキシングとして、大規模木構造データベースから類似木を効率良く取り出すためのインデキシングとして、メトリック空間に基づいた方法を検討した。またテキスト情報に基づいたインデキシングとして、ジャカード係数を用いたインデキシングを試みた。

①メトリック空間検索

これまでに研究が進められてきたメトリック空間での検索技術を応用するために、まず、木構造データを多次元ベクトルで表す方法を検討した。本研究では、研究代表者等が以前の研究で提案した部分木を用いた特徴ベクトルを用いた。この特徴ベクトルは、木に含まれるすべての部分木を特徴として用いるものである。図４は、２つの木、T_1 と T_2 から得られる特徴を表している。この例では、２つのラベル a,b が含まれているため、１つのノードからなる部分木は２種類になる。同様に、２つの木に含まれる、各大きさの部分木が図の t 行に列挙されている。各部分木で表される特徴に対して、その出現回数を特徴

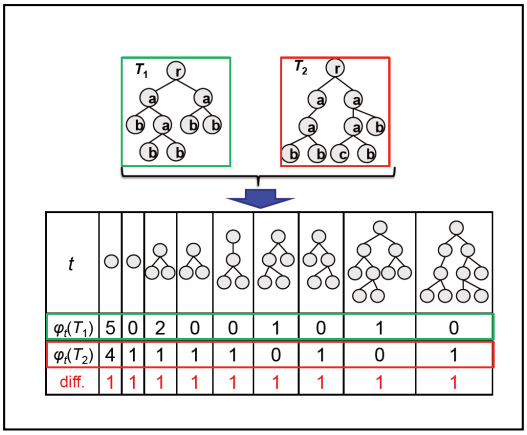

図 4 無順序木の特徴ベクトル

量として考える。たとえば、T_1 には a のラベルをもつノードが５個含まれているため、ラベル a のついた１つのノードからなる部分木に対する特徴量は 5 となる。図４の $\varphi_t(T_1)$ と $\varphi_t(T_2)$は、このような方法で得られた木 T_1 および T_2 の特徴ベクトルを表している。

次に、上記の方法で得られた木の特徴ベクトルの L1 距離を考える。図 4 の diff 行は、各特徴量の差を示しており、その和である９が T_1 と T_2 の L１距離となる。木の編集距離の間には式２の関係が成り立つ。式２において h は木の高さを表している。また、$||\cdot||_1$ は、L1 距離を示している。

$$\frac{1}{2h+2}\left\|\varphi(T_1)-\varphi(T_2)\right\|_1 \le TED(T_1,T_2) \le \left\|\varphi(T_1)-\varphi(T_2)\right\|_1$$

数式 2　木編集距離の上下限

検索においては、データベース中の各木を上記の特徴ベクトルに変換し、L1 距離に基づいて、問い合わせに類似した木を絞り込む。ここで、メトリック空間検索で用いられるピボットを用いた空間の分割を行う。ピボットの選択法には、空間の端に位置するピボットを選択するなどのヒューリスティックな方法が提案されてきた。本研究では、ピボットによって分割される部分空間の間のマージンが大きくなるように空間を分割する方法を用いた。図５は、その概略を表している。ピボットとして図５右下に位置する P を使う場合に、P とデータベース中の木（図中の円）の L１距離を計算する。その距離に基づいて、木のクラスタを作成し、クラスタ間の距離が最大になるように空間を分割する方法である。これによって、高次元の特徴ベクトルに対する検索処理効率の向上を図った。

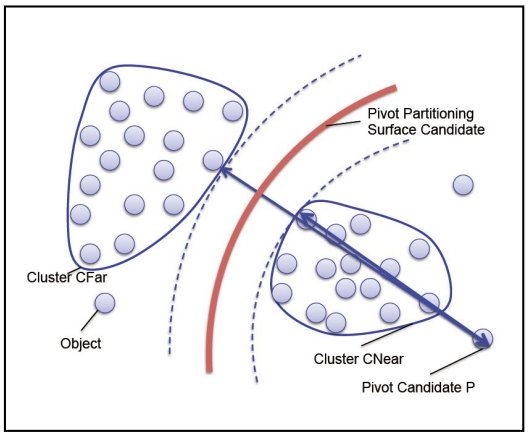

図 5　マージン最大空間分割

②テキスト情報を用いた検索

テキスト情報に着目した方法として、木に付与されているテキストのジャカード係数に基づいたフィルタリングを行った。まず、各木に付与されているテキストから単語を抽出し、木の特徴をその頻度ベクトルで表す。次に頻度ベクトル間のジャカード係数を類似度する similarity join を行い、類似木のクラスタを求める。最後にクラスタ内のすべての木のペアの類似度をはかることによって、現実的な時間内で木のマッチングを行う。

本研究では、学術論文の書誌データベースである DBLP と ACM の電子図書館、Google Scholar に含まれる書誌レコードの統合実験を行った。評価データとして、Leizig 大学が提供する情報統合評価用コーパスを用いた。表２は評価用コーパスに含まれているデータの大きさを示している。書誌統合を行う場合は、各データベースのレコードのすべての組みについて類似度を計算することになるため、膨大な数のマッチング処理が必要になる。そこで、上記の similarity join を用いて候補レコードペアを表中の blocking result の行に示される数まで絞り込んだ。

	ACM-DBLP	Scholar-DBLP
pattern articles	2,616	2,616
DBLP articles	2,294	64,263
common articles	2,224	5,347
blocking result	4,000	20,000

表 2　データベースサイズ

次に、各候補レコードペアに対して、本研究で考案した無順序木のマッチングアルゴリズムを適用し、各ペアの類似度を計算した。類似度の計算の過程で、木のノード間の対応づけも行われる。そこで、このマッピングの結果を用いて、２つのデータベースのスキーマのマッチングを行った。

表３は DBLP と ACM, Google Scholar の書誌データのスキーマのマッチングを行った結果を示している。この実験では、書誌レコードの著者、タイトル、出版年などの書誌要素に対応するタグ間のマッチング精度を調べた。例えば、表３(a)の author 行、０列の数値 7962 は、DBLP に含まれる書誌レコード中の 7962 名の著者は、すべて ACM 電子図書館の著者フィールドとマッチングできたことを示している。また、DBLP の開催都市フィールドのうち、71 件は著者に 148 件はタイトルフィールドとマッチしている。これは、評価用データセットの ACM 図書館データに開催都市に関する情報が含まれていなかったことが主な原因と考えられる。この表からわかるように主要な書誌要素に対応するタグについては、２つのデータベース間で正しく対応づけられており、無順序木のマッチングは、XML データに対して、高いマッチング能力を有していることがわかった。

(a) ACM-DBLP

	0 (article)	1 (authors)	2 (author)	3 (title)	4 (year)
article	**3893**	0	0	0	0
authors	0	**3672**	0	0	0
author	0	0	**7962**	0	0
title	0	0	0	**3378**	0
year	0	0	0	0	**2987**
url	0	10	0	283	529
cites	0	95	0	44	338
ee	0	71	0	148	102
cite	0	0	199	0	0
others	0	45	126	40	37

(b) Scholar-DBLP

	0	1	2	3	4
article	**18475**	0	0	0	0
authors	0	**13519**	0	0	0
author	0	0	**27217**	0	0
title	0	0	0	**15680**	0
year	0	0	0	0	**5082**
url	0	0	0	1167	2772
cite	0	0	5768	0	0
editors	0	1798	0	0	0
editor	0	0	3365	0	0
others	0	3158	0	1261	2334

表 3　スキーママッチング結果

５．主な発表論文等

（研究代表者、研究分担者及び連携研究者には下線）

〔雑誌論文〕（計 2 件）

① Tatsuya Akutsu, Takeyuki Tamura, Avraham A. Melkman, Atsuhiro Takasu. On the Complexity of Finding a Largest Common Subtree of Bounded Degree. Lecture Note in Computer Science 8070, pp.4 - 15, 2013 [査読有], DOI: 10.1007/978-3-642-40164-0_4.

② Tatsuya Akutsu, Takeyuki Tamura, Daiji Fukagawa, Atsuhiro Takasu. Efficient Exponential Time Algorithms for Edit Distance between Unordered Trees. Lecture Note in Computer Science 7354, pp.360 - 372, 2012 [査読有], DOI: 10.1007/978-3-642-31265-6_29.

〔学会発表〕（計 4 件）

① Quong-Hong Vuong, Atsuhiro Takasu. K Transfer Learning for Bibliographic Information Extraction. in Proc. of ICPRAM2015, pp.374 - 379, January 12, 2015. [poster, 査読有], Lisbon(Portugal)

② Tatsuya Akutsu, Jesper Jansson, Atsuhiro Takasu, Takeyuki Tamura. On the Parameterized Complexity of Associative and Commutative Unification. in Proc. of IPEC2014, pp.15 - 27, September 12, 2014. [full paper、査読有], Wroctaw(Poland)

③ Quong-Hong Vuong, Atsuhiro Takasu. K Transfer Learning for Emotional Polarity Classification. in Proc. of WI2014, pp.94 - 101, August 13, 2014. [full paper,査読有], Warsaw(Poland)

④ 大橋駿介，高須淳宏，相澤彰子．表記が異なる同義数式の高速な検索法．第６回データ工学と情報マネジメントに関するフォーラム，2014 年 3 月 3 日、[査読有]，ウェスティン淡路（淡路市、兵庫県）．

〔その他〕

受賞

① 学生プレゼンテーション賞 at 第６回データ工学と情報マネジメントに関するフォーラム（DEIM 2014），表記が異なる同義数式の高速な検索法，大橋駿介，相澤彰子， 高須淳宏，2014.3.

６．研究組織

(1)研究代表者

高須　淳宏（TAKASU, Atsuhiro）
国立情報学研究所・コンテンツ科学研究系・教授
研究者番号：90216648