

構造型 P2P ネットワークにおけるキーワードを用いた XML 文書検索

李曉晨[†]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[‡] 筑波大学計算科学研究センター

〒305-8573 つくば市天王台1-1-1

1. はじめに

Peer-to-Peer システムとは、参加するコンピュータ (peer) が対等の立場でネットワークを構成するシステムのことであり、オーバレイネットワークの基盤技術として近年注目されている。P2P システムの一手法として、分散ハッシュテーブル (DHT) がある。他方、XML はタグで文書やデータの意味や構造を記述するためのメタ言語であり、標準的なデータフォーマットとして広く利用されている。オーバレイネットワークにおいても、データやデータに付随するメタデータのフォーマットとして利用されることが予想される。

現在、XML 文書の検索には XPath や XQuery などのパスに基づいた問合せ言語を用いることが多い。しかし、そのためには問合せ言語に対する専門的知識や、検索対象の XML 文書の構造をあらかじめ把握する必要がある。しかし、実際の P2P 環境において異なる利用者がさまざまな XML スキーマを用いる可能性があり、全ての XML 文書構造を把握することは現実的ではない。

そこで、我々はP2P環境におけるXMLデータのキーワード検索のため、DHTに基づいた転置リストの実現手法を提案する。これにより、検索キーワードになるべく強い関連を持つXML部分文書を検索できる。基本的には、XMLデータを葉ノード (テキストノード) と属性ノードに分解してDHTに格納する。検索時には、DHTによりキーワード情報を検索して、ピア間でSLCA[2]

(Smallest Lowest Common Ancestors) を計算する。データ量が多くなるとピア間での通信をするためのトラフィックが大量に発生してしまうという問題があるため、本研究では、Bloom Filterという集合要素の圧縮手法を用いて、ピア間での通信トラフィックを削減する方式を提案する。

2. 基本的事項

2.1 分散ハッシュ表 (DHT)

DHT では、システムで共通のハッシュ関数を用意し、このハッシュ関数から得られる値を元に、オブジェクト (ピア、データ) をオーバレイネットワーク上に配

置する。DHTには、代表的な方式として、Chord[1]、CANなどがある。以下では本研究で用いるChord について紹介する。

Chord では、 2^N 個のオブジェクトからなる円構造のハッシュ空間 (ID サークル) を用いる。N はスケールファクタと呼ばれる。オブジェクトはハッシュ関数でID サークル上にマップされる。ID_i のピアは、担当するデータのID、経路表、前後に位置するピアのID 情報(predecessor, successor) を保持している。またピアi が担当するデータのID は、predecessorからi までの区間である。経路表には、N 個の他ピアへのリンクがあり、リンク先は $i+2^k$ ($k=0, 1, 2, \dots, N-1$) 位置の担当ピアである。このため、検索に要するホップ数は $O(\log 2^N)$ である。

2.2 SLCA (Smallest Lowest Common Ancestors)

XML は木構造を有するため、キーワード検索の際には、キーワードにマッチする部分文書を決定する必要がある。このためにSLCA という概念を用いる。まず、最小共通祖先 LCA (Lowest Common Ancestors) とは、木においていくつかのノードの共通の祖先の内、最も葉に近いノードである。一般に、あるキーワード集合にマッチするノードのLCA は複数存在するが、その中で最小の (部分木にLCA を含まないような) LCA をSLCA (Smallest LCA) と呼ぶ。

3. 提案手法

3.1 KW-DHT による XML データの格納

本研究では、キーワード検索を扱う。このため、提案する転置リストでは、XML データのテキスト値と属性値のみについて構築する。すなわち、XML データの葉ノード (テキストノード) と属性ノードのみをDHTに格納する。このためのDHTをKW-DHTと呼ぶ。具体的には、XML データの葉ノード (テキストノード) と属性ノードを抽出し、その中の各単語をハッシュキー、単語が含まれるノードの要素名、LocationID の組を値とする。LocationID は、単語が出現する場所の位置情報を保持するIDであり、ホストID、文書ID、DeweyIDの連結である。

3.2 KW-DHT によるキーワード検索

P2P ネットワークの利用者が検索する際、余計な情

Keyword Search on XML Document in Structured P2P Network

†Xiaochen LI University of Tsukuba

‡Toshiyuki AMAGASA University of Tsukuba

‡Hiroyuki KITAGAWA University of Tsukuba

報を含まずに検索キーワードと関連の強い最小部分文書を求めたいので、検索キーワードを含んだ SLCA を解とする。

3. 2. 1 KW-DHT における SLCA の算出方法

P2P ネットワーク上では、キーワードの転置リストが分散配置されている。与えられたキーワード集合を含む部分文書から SLCA を計算するには、LocationID をピア間でやり取りする必要がある。図1に例を示す。まず、各検索キーワードを含む LocationID のリストのサイズの小さいものから順に処理を行う。最初のピアにおいて、検索キーワードの LocationID リストを検索し、次のピアに送信する。次のピアは、送られた LocationID リストと自分が保持する（キーワード集合にマッチする）LocationID リストから SLCA を計算する。これを全てのキーワードについて行うことで、全キーワードを含む SLCA を計算することができる。しかし、複数のキーワードによる SLCA の算出を行う際、多くの LocationID を送信するためのトラフィックが大量に発生してしまうという問題がある。

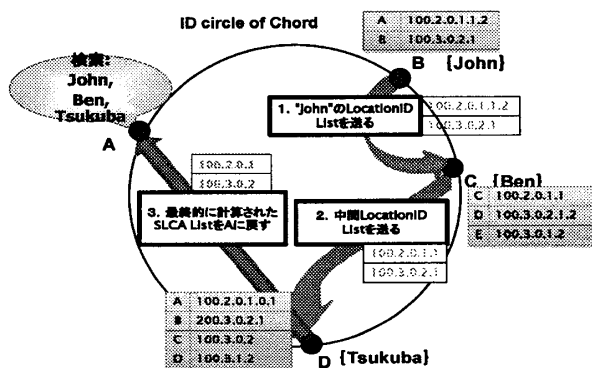


図1: KW-DHT を利用した検索処理の例

4. Bloom Filter による通信速度の改善

上で指摘した問題点を克服するため、本研究では Bloom Filter を用いて送信データの圧縮を試みる。

4. 1 Bloom Filter

Bloom Filter とは、複数の要素から構成される集合の中に、任意の要素が含まれているかどうかを判定するアルゴリズムである。キーワードはビットデータとして格納されているため、ノード間で情報を交換する際は少ない通信負荷で済む。

4. 2 適用手法

Bloom Filter の適用について、二つ問題点がある。

1. SLCA を算出するため、Bloom Filter をどのように構成するか。
2. Bloom Filter では、ハッシュ関数を用いるため、

ハッシュ関数による誤検知 (False Positive) が存在する。

4. 2. 1 Bloom Filterの構成

XML データは木構造を有するため、ノード間の先祖・子孫関係を考慮して SLCA を算出しなければならない。このため、各 LocationID だけではなく、それらの全ての先祖ノードを Bloom Filter に格納する。例えば、LocationID が 100. 2. 0. 1. 1. 2 とすると、0. 1. 1. 2 による親を抽出して、Bloom Filter に集合 (100. 2. 0, 100. 2. 0. 1, 100. 2. 0. 1. 1, 100. 2. 0. 1. 1. 2) を入れる。これにより、先祖・子孫関係を考慮した SLCA のマッチングが可能となる。

4. 2. 2 誤検出の解消

続いて、Bloom Filter の誤検出の解消について述べる。基本的なアイデアは、Bloom Filter と実際の LocationID リストを組み合わせることである。まず、Bloom Filter のみで SLCA を計算し、(誤りを含む) SLCA を求める。いったん求めた SLCA を今度は全てのピアに再度送信し、各ピアで論理和を計算することで、誤検出した LocationID を排除する。

そこで、SLCA を算出した後は一般的に LocationID の数が前より少なくなるため、二回目に周る際、伝送される LocationID のサイズがかなり小さいことが期待できる。よって、全体の検索効率を上げることができる。

5. まとめと今後の課題

本研究では、構造型P2P環境でキーワードを用いる XML 文書の格納・検索手法を提案した。XML データのテキスト単語をハッシュキーとして KW-DHT に格納する。ピア間で KW-DHT による検索キーワードの SLCA を算出する。最後に、Bloom Filter を用いて、通信速度を改善する。今後は、システムの実装を行い、実験による評価を行う予定である。

謝辞

本研究の一部は科学研究費補助金若手研究 (#19700083)、科学技術振興機構 CREST「自律連合型基盤システムの構築」による。

参考文献

- [1] Stoica, I., Morris, R., Karger, D., Kaashoek, M.F., Balakrishnan, H. Chord: A scalable peer-to-peer lookup service for Internet applications. In Proc. ACM SIGCOMM'01, San Diego, CA, Aug. 2001.
- [2] Y. Xu and Y. Papakonstantinou. Efficient Keyword Search for Smallest LCAs in XML Databases. In Proceedings of SIGMOD, New York, NY, USA, 2005.