

並列相関関係抽出実行時の IP-SAN 統合型 PC クラスタの特性評価

原 明日香[†] 神坂 紀久子[†] 小 口 正 人[†]

近年発達する情報化社会において、データマイニングが注目されており、その中に最も処理が重いとされる相関関係抽出という問題がある。その解法として Apriori アルゴリズムと FP-growth アルゴリズムが有名である。また、処理の対象となるデータは大規模なものが多く並列化が不可欠となるため、フロントエンドとバックエンドのネットワークを同じ IP ネットワークに統合した IP-SAN 統合型 PC クラスタを提案した。その上で 2 つの並列化アルゴリズムを動作させ、比較を行うとともに、IP-SAN 統合型 PC クラスタの特性評価を行った。

Evaluation of characteristics of IP-SAN consolidated PC cluster when parallel association rule mining is executed

ASUKA HARA,[†] KIKUKO KAMISAKA[†] and MASATO OGUCHI[†]

In the information society that develops in recent years, data mining is attracts much attention, and one of data mining problems called association rule mining seems to be the heaviest one to process. The Apriori algorithm and the FP-growth algorithm are famous as the technique to solve it. Moreover, it is essential to execute parallel processing because the target data is enormous in many cases. Thus, we propose IP-SAN consolidated PC cluster that integrated the network of the front-end and the back-end into the same IP network. Two parallel algorithms were executed on that and compared, and the characteristics of IP-SAN consolidated PC cluster was executed.

1. はじめに

近年、発達する情報化社会では、データの蓄積と運用が非常に重要になってきている。また、情報システムにおいて処理されるデータ量が膨大になってきている。ユーザにとって重要なデータが蓄積されているにも関わらず、使いこなせていない場合が少なくない。そこでデータマイニングの中で、膨大なデータから有益な規則や関係を抽出する相関関係抽出に注目した。相関関係抽出のためのアルゴリズムとして代表的なものに、候補アイテムセットから頻出アイテムセットを抽出する動作を繰り返すことで相関関係を抽出する Apriori アルゴリズムと、巨大なトランザクションデータベースから相関関係抽出に必要な情報をコンパクトに圧縮したデータ構造である FP-tree と呼ばれるものを利用することにより候補パターンを生成することなく頻出パターンを抽出する FP-growth アルゴリズムがある。

パラメータの条件によっては相関関係抽出における

計算量、データ処理量は非常に多くなるため、並列化が不可欠となる。そのため分散メモリ型並列計算機である PC クラスタを用いるが、その際に我々が提案している Front-end と Back-end のネットワークを同じ IP ネットワークに統合した IP-SAN 統合型 PC クラスタを用いる。相関関係抽出の並列化アルゴリズムとして HPA (Hash Partitioned Apriori) と HPA を元に提案された PFP (Parallelized FP-growth) を使用する。

HPA アルゴリズムと PFP アルゴリズムを、ローカルデバイスを用いた PC クラスタ、IP-SAN 統合型 PC クラスタ、IP-SAN 非統合型 PC クラスタ上で実行し、実行時間を測定したところ PFP アルゴリズムの方が格段に速いということ、どの PC クラスタも同程度の性能であるということが分かった。しかし現状の実験環境では PFP アルゴリズムはデータの量が大きくなると動作しなかったため、その原因を解明し、より大規模なデータマイニングを行えるようにプログラムの修正を行い、動作させたときの IP-SAN 統合型 PC クラスタの特性評価を行う。

[†] お茶の水女子大学
Ochanomizu University

2. IP-SAN 統合型 PC クラスタ

各ノードが独立して動作する CPU、メモリ、二次記憶を保有し、ノードが必要に応じてネットワークを介し互いに通信することで全体として並列分散処理を実現する分散メモリ型並列計算機の各ノードに汎用のパーソナルコンピュータとネットワークを用いたものを PC クラスタ [図 1] という。Front-end(ノード間通信)は LAN(Local Area Network)、Back-end(ストレージアクセス)は SAN(Storage Area Network)でネットワーク接続されている。

そこで図 2 のように Front-end と Back-end のネットワークを同じ IP ネットワークに統合した IP-SAN 統合型 PC クラスタの実現を考えた。ネットワークを統合することでネットワーク構築コストと管理コストの削減が可能となる。

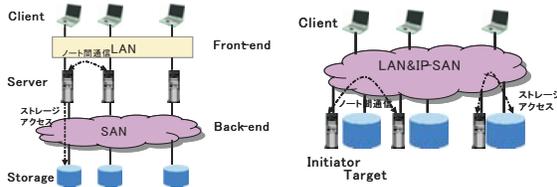


図 1 通常の PC クラスタ 図 2 IP-SAN 統合型 PC クラスタ

3. 相関関係抽出とその並列化

相関関係抽出では、巨大なデータから有益な規則性や関係を抽出するためにあるパターンが現れる頻度(サポート値)を調べる。その頻度が多ければ、そのパターンから得られる関係は有意義なデータとなり、販売戦略などに活用出来る。

相関関係抽出で扱うデータはしばしば巨大であるため、データベースを分散し計算処理を並列化して、多数台のコンピュータをネットワークで接続した PC クラスタなどの環境でマイニング処理を実行する並列相関関係抽出の研究が行われている [1]。以下に相関関係抽出の代表的な 2 つのアルゴリズムの概要を説明し、本研究で用いる並列化アルゴリズムを紹介する。

3.1 Apriori アルゴリズム

1994 年に Agrawal らによって提案されたもので、発見された頻出アイテムセットから候補アイテムセットを生成し、繰り返し数え上げを行っていくアルゴリズムである [3]。まず 1 回目のデータベーススキャンで長さ 1 の頻出 1 アイテムセットを抽出し、それらを元に長さ 2 の候補 2 アイテムセットを生成する。次に

2 回目のデータベーススキャンで候補 2 アイテムセットから頻出 2 アイテムセットを抽出する。これを候補アイテムセットが生成されなくなるまで繰り返していくことで、頻出パターンをすべて発見していくアルゴリズムである。

Apriori アルゴリズムには、候補アイテムセットを格納するために大容量のメモリが必要となる、何度も繰り返しデータベースをスキャンする可能性があるといった問題点がある。

Apriori をベースにした並列相関関係抽出のアルゴリズムはいくつか提案されているが、本研究ではハッシュ関数を使用して Apriori を並列化する HPA (Hash Partitioned Apriori) [2] を用いる。

3.2 FP-growth アルゴリズム

2000 年に Han らによって提案されたもので、巨大なトランザクションデータベースから相関関係抽出に必要な情報をコンパクトに圧縮したデータ構造である FP-tree を利用している [4]。候補パターンを生成せずに頻出パターンを抽出することで、Apriori アルゴリズムの問題点を改善したアルゴリズムである。

FP-tree は次のように構築される。1 回目のデータベーススキャンで、各アイテムのサポート値を求め、頻出アイテムを抽出する。抽出された頻出アイテムをサポート値により、頻度が降順になるように並び替え(これを F-list とする)、空 (null) のラベルを持つ木のルートを作成(これを T とする)する。2 回目のデータベーススキャンで、F-list に従ってトランザクションから頻出アイテムを抽出し、ソートする。T が F-list の要素である子を持っていれば、その子のカウントを 1 増やし、持っていなければ、新しくカウント 1 を持つ子を作る。F-list の最後までこの操作を繰り返す。FP-tree の構築例を図 3 に示す。

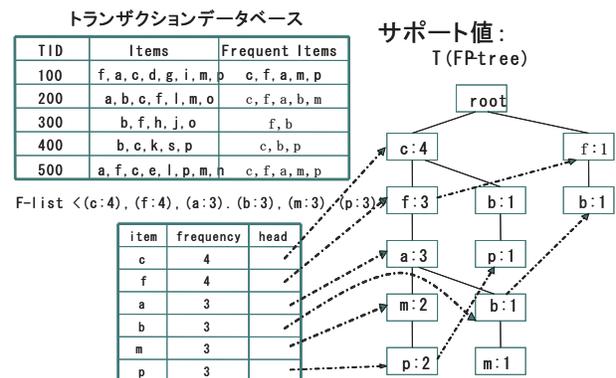


図 3 FP-tree の構築例

FP-growth は構築された FP-tree の性質を利用することにより、頻出パターンを発見していくアルゴリズムである。FP-growth の並列相関関係抽出のアルゴリズムは、本研究では HPA を元に行われた既存研究 [4] で提案された PFP(Parallelized FP-growth) を用いる。

4. 実験内容と実験結果

4.1 既存研究

我々は、HPA アルゴリズムと PFP アルゴリズムの並列化プログラムについて、アイテム数を 1000 とし、トランザクション数が 10K、20K、40K、80K、160K のトランザクションデータを用い、最小サポート値を 0.7 % として実行した。プラットフォームとしては、ローカルデバイス (SCSI ディスク) を用いた PC クラスタ、IP-SAN 統合型 PC クラスタ、バックエンド IP-SAN を用いた非統合型 PC クラスタで実行し、そのときの実行時間をそれぞれ測定した。実験には 4 台の PC を Gigabit Ethernet で接続した PC クラスタを用いる。IP-SAN を用いる場合には、iSCSI ターゲット用の PC がもう 4 台接続されている。各 PC は CPU が Pentium4 1.5GHz、メインメモリが 384MB、OS が Linux 2.6.9-1667(Fedora core 3) である。

図 4 に同じ大きさのトランザクションデータにおける 2 つのアルゴリズムの実行時間を示す。明らかに PFP アルゴリズムの方が速いことがわかる。これは、HPA アルゴリズムが頻出アイテムセットから候補アイテムセットを作るといった動作を、データの大きさに関係なく何度も繰り返し行っているためである。反対に PFP アルゴリズムはデータの大きさによって、FP-tree の大きさが変わるため、データ量が比較的小さい今回の実験では、良い結果が出たと考えられる。

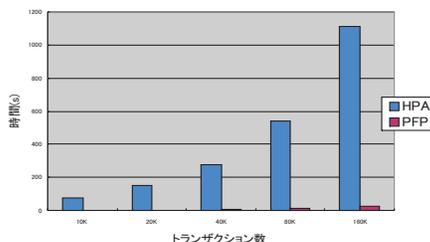


図 4 iSCSI 統合型 PC クラスタにおける各アルゴリズムの実行結果

図 5 に HPA アルゴリズム、図 6 に PFP アルゴリ

ズムのそれぞれのクラスタにおける実行時間を示す。どのクラスタにおいても実行時間はほとんど変わらなかった。IP-SAN 統合型 PC クラスタは、バックエンドとフロントエンドで同じネットワークを使用するため性能が落ちる可能性が予想されたが、本実験で性能は変わらないということが分かった。

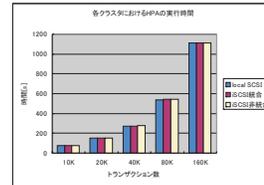


図 5 各クラスタにおける HPA アルゴリズムの実行結果

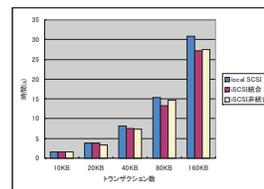


図 6 各クラスタにおける PFP アルゴリズムの実行結果

4.2 本研究における実験

そこで既存研究をふまえ、本研究ではより大規模なデータにおける PFP アルゴリズムの並列化プログラムの動作を解析し、その改良を行った。

既存研究で使用していた PFP 並列化プログラムは、非頻出アイテムなどに多くの無駄なメモリが使われてしまっているため、確保されたメモリ容量を超えてしまうということが分かった。そこで、既存研究で使用していた PFP アルゴリズムの並列化プログラムの最適化を目指すことを目的とし、非頻出アイテム集合を含むメモリの圧縮などを行うことでメモリ管理を充実させたプログラムを本研究の実験環境に移殖し、実行時間を測定した。アイテム数を 1000 とし、トランザクション数が 1M、2M、4M、8M、16M のトランザクションデータを用い、最小サポート値を 0.7 % として、ローカルデバイス (SCSI ディスク) を用いた PC クラスタ、IP-SAN 統合型 PC クラスタで実行し、そのときの実行時間をそれぞれ測定した。

図 7 に同じ大きさのトランザクションデータにおける 2 つのアルゴリズムの実行時間を示す。既存研究と同様に明らかに PFP アルゴリズムの方が速いことがわかる。既存研究では PFP アルゴリズムは HPA アルゴリズムの約 50 倍の性能となったが、本研究では約 10 倍となり、性能は劣ったが大規模なデータを扱

うことができるようになった。

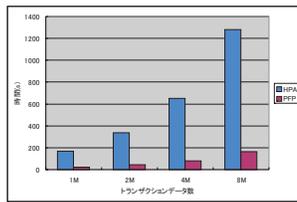


図7 iSCSI 統合型 PC クラスタにおける各アルゴリズムの実行結果

図8にHPAアルゴリズム、図9にPFPアルゴリズムのそれぞれのクラスタにおける実行時間を示す。どのクラスタにおいても実行時間はほとんど変わらなかった。IP-SAN 統合型 PC クラスタは、バックエンドとフロントエンドで同じネットワークを使用するため性能が落ちる可能性が予想されたが、性能は変わらないということが分かった。トランザクションデータ数が多い場合にはむしろ IP-SAN 統合型 PC クラスタの方が良い結果になった。この理由については、今後検討を進める。

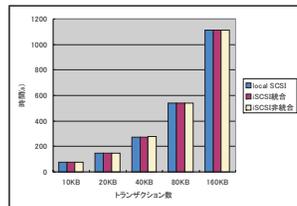


図8 各クラスタにおける HPA アルゴリズムの実行結果

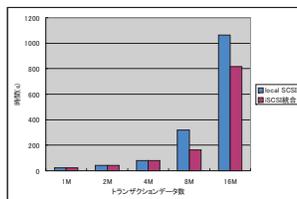


図9 各クラスタにおける PFP アルゴリズムの実行結果

5. まとめと今後の予定

本研究では、Aprioriの並列化アルゴリズム HPA と FP-growth アルゴリズムの並列化アルゴリズム PFP を、これまでより大規模なデータを用いて、ローカルデバイスを用いた PC クラスタ、IP-SAN 統合型 PC クラスタ、バックエンド IP-SAN を用いた非統合型 PC クラスタにおいて実行し、実行時間を測定した。

これまで用いていた PFP アルゴリズムの並列化プログラムの性質上、小規模なデータを用いた場合、HPA アルゴリズムと PFP アルゴリズムの実行時間を比較すると、PFP アルゴリズムの方が約 10 倍速かったが、本研究で PFP アルゴリズムのメモリ管理を改善したところ、約 10 倍 PFP アルゴリズムが速いという結果になった。したがって、本研究室の実験環境においてもデータの大きさに関係なく PFP アルゴリズムが有効であるということが分かった。

また、ローカルデバイスを用いた PC クラスタと IP-SAN 統合型 PC クラスタの比較では、これまでと同様にどちらのアルゴリズムにおいても、3 つの PC クラスタにおける実行結果はほとんど変わらなかった。データ量が多くても、コストが安く、ネットワークの管理のしやすい IP-SAN 統合型 PC クラスタが有効であると考えられる。

今後は PC クラスタ数を増やし、より大規模なデータをより速くマイニングしていきたいと考えている。対象データとして今回用いた人工的に作られたデータではなく、バイオインフォマティクスなどのこれから重要となると考えられるデータを用いて実験を行っていききたい。また、IP-SAN 統合型 PC クラスタの最適化なども行っていききたい。

参考文献

- 1) 福田剛志、森本康彦、徳山剛志:”データマイニング”, 共立出版
- 2) 小口正人、喜連川優:”ATM 結合 PC クラスタにおける動的リモートメモリ利用方式を用いた並列データマイニングの実行”, 電子情報通信学会論文誌, Vol. J84-D-I, No.9, pp.1336-1349, 2001 年 9 月
- 3) R. Agrawal, T. Imielinski, A. Swami:”Mining Association Rules Mining between Sets of Items in Large Databases”, In Proceedings of the International Conference on Very Large Data Bases, pp.487-499, 1994.
- 4) J. Han, J. Pei, and Y. Yin:”Mining Frequent Patterns without Candidate Generation”, ACM SIGMOD2000, pp.1-12, May 2000
- 5) Iko Pramudiono and Masaru Ksuregawa:”Tree structure based Parallel Frequent Pattern Mining on PC cluster”, DEXA2003, pp.537-539, September 2003
- 6) 原明日香、神坂紀久子、小口正人:”IP-SAN 統合型 PC クラスタにおける相関関係抽出の実行”, 第 69 回情報処理学会全国大会, 5S-5, 2007 年 3 月発表