社団法人 情報処理学会 研究報告
IPSJ SIG Technical Reports

2006－ＤＳＭ－41（６）
2006／5／11

プライバシーを保護した分散データマイニングアルゴリズムに関する研究

浦邊 信太郎　王 家宏　児玉 英一郎　高田 豊雄

岩手県立大学 ソフトウェア情報学研究科

〒020-0193 岩手県岩手郡滝沢村滝沢字巣子 152 番地 52

E-mail: g231d003@edu.soft.iwate-pu.ac.jp, {wjh, kodama, takata}@iwate-pu.ac.jp

あらまし ネットワーク上に分散されたデータベースから共通の相関ルールを発見する分散データマイニングにおいて，重要な問題の１つにプライバシーの保護がある．例えば，異なるクレジットカード会社同士で共通のカード詐欺のパターンを抽出する際，データマイニングを行うためには自社のカード詐欺事例を他社へ伝える必要があるが，自社の契約内容などを知られることは不利益につながる．このような問題を解決するため，本研究では，他サイトに自分のデータを知られることなくデータマイニングを行い，結果のみを共有できる分散データマイニングアルゴリズムを提案する．

キーワード データマイニング，プライバシー保護，分散処理

A Collusion-Resistant Approach to Distributed Privacy-Preserving Data Mining

Shintaro URABE, Jiahong WANG, Eiichiro KODAMA, and Toyoo TAKATA

Faculty of Software and Information science, Iwate Prefectural University

152-52 Sugo, Takizawa, Iwate 020-0193, Japan

E-mail: g231d003@edu.soft.iwate-pu.ac.jp, {wjh, kodama, takata}@iwate-pu.ac.jp

Abstract It would be profitable to every member of a group of companies or organizations for them to conduct a data mining to discover the shared patterns, associations, trends or dependencies in their shared data. The privacy, however, is a concern. It is often required that the desired results of data mining are accomplished while still preserving the privacy of every participating group member. In this paper, a distributed privacy-preserving data mining algorithm is proposed. The algorithm is characterized by its excellent collusion-resistant ability, which is achieved in very low communication cost.

Keyword Data mining, preserving privacy, distributed processing, collusion-resistant communication.

1. はじめに

近年，企業で取り扱う様々なデータが電子化され，データベースの利用がより一般的になっている．それに伴い，ネットワーク上に分散されたデータベースからビジネスに活用できる有用な情報を取り出す技術として分散データマイニングが注目されている．

既存の分散データマイニングアルゴリズムにFDM[1]というアルゴリズムがある．しかし，FDMは自サイトのデータをそのまま他のサイトへ送信する必要があり，プライバシーが漏洩する可能性がある．そのため，分散データマイニングにおけるプライバシー保護のための手法が提案されている．ここでプライバシー保護とは，あるサイトの持つ「トランザクション」及び「興味深い相関ルール」が他のサイトに漏洩するのを防ぐことと位置づける．SFDM[2]は暗号化を用いてデータを秘匿し，プライバシーを保護する手法の１つである．しかし，暗号化を用いるため計算機に大きな処理負荷がかかってしまうという問題がある．SDDM[3]は乱数を用いることによってこの問題を改善している．しかし，SDDMは高々４つのサイトが結託することによって情報が漏洩してしまうという問題がある．そこで本研究ではサイト間の結託に対する耐性を向上させる分散データマイニングアルゴリズムを提案する．本提案アルゴリズムでは乱数を複数用いること

によって情報を集計するルートを複数生成し，サイト間の結託に対する耐性を向上させる．本稿ではプライバシーを保護した分散データマイニングアルゴリズムを示すと共に，本提案アルゴリズムが既存のアルゴリズムと比較してサイト間の結託に対する耐性が高いことを示す．

2. システムモデル

本研究で対象とするのは，ネットワークに接続された複数のコンピュータが同じスキーマのデータベースを保持し，それぞれでデータマイニングを行い，結果を統合するシステムである．このデータマイニングから導き出される「X を買う人は Y も一緒に買う」のようなアイテム間の関係をアソシエーションルールと呼び，X⇒Y と表す．ここで，X, Y はそれぞれアイテムセットを表す．アイテムセットとは，1 つ以上のアイテムで構成される集合である．

アソシエーションルール X⇒Y はサポート(support)と確信度(confidence)という 2 つの評価尺度によってその有効性が計られる．サポートはデータベースの中でアイテムセット X と Y を含むトランザクションの存在する割合を表し，確信度は X を含むトランザクションが Y も含む条件付き確率を表す．なお，ここで言うトランザクションとは顧客が購入した商品の履歴等であり，データベースとはトランザクションの集合である．データマイニングのユーザは，サポートの「下限値」(ユーザが任意に設定)を超える「興味深いアイテムセット」だけを必要とする場合を考える．サポートの下限値を min_sup, データマイニングを行うサイトの数を M とし，i 番目のサイトでアイテムセット X の出現するトランザクションの数を X_i, データベースのサイズを d_i,とすると以下の式が成り立つアイテムセット X は頻出なアイテムセットとなる．

$$\sum_{i=1}^{M}(X_i - \text{min\_sup} * d_i) \geqq 0$$

3. 提案するアルゴリズム

本研究で提案するアルゴリズムの目的は，頻出なアイテムセットやサポートを集計する際に各サイトがそれぞれ発信する値を秘匿することである．そこで，各サイトでは値を発信する際に乱数を付与することで秘匿を行う．続いて，各サイトで付与した乱数を集計し，乱数が付与された値の合計から乱数の合計を減算することによって値の合計を得る．以上の手順を用いることによって，各サイトで発信した値を秘匿しつつ値の合計のみを得ることができる．

Algorithm: データ提供者を秘匿することによりプライバシー漏洩を防ぐ分散データマイニング

Input: サポートと確信度の下限値，各サイトにあるデータベースからなる集合

Output: 全ての興味深いアソシエーションルール

Step1: 起点となるサイトが候補アイテムセットを全サイトに送信する．

Step2: 各サイトは候補アイテムセットを興味深いか否か判定し，評価値（興味深ければ値 1，興味深くなければ値 0）を決定する．

Step3: *Step2* で決定した評価値を全体で秘匿したまま集計する．

Step4: 評価値の合計が 1 以上であれば，興味深い可能性があると判断し，その候補アイテムセットを全サイトにブロードキャストする．

Step5: ローカルで候補アイテムセットのサポートを導出し，全体で秘匿したまま集計する．

Step6: サポートの下限値を上回る候補から確信度の下限値を上回る候補を選択し，興味深いアソシエーションルールとして出力する．

上記のアルゴリズムを使用することにより，各サイトの興味深いアイテムセットやサポートを他のサイトに知られることなくデータマイニングを行うことができる．実際に評価値及びサポートを秘匿しながら集計を行っているのは Step3 と Step5 である．Step3 と Step5 では，集計を行う際にそれぞれのサイトで発信する値を秘匿している．具体的な集計アルゴリズムを次ページ左に示す．

この集計アルゴリズムを使用することで，各サイトの集計するべき値 V_i を秘匿しながらその合計のみを得ることができる．このアルゴリズムでは集計セッションという手続き(次ページ右)を複数回行う．これは秘匿すべき値やそれを秘匿する乱数などを個別に集計する手続きである．

Algorithm: 各サイトの値を秘匿しながら集計を行うアルゴリズム

Input: (1) M: サイト数
(2) V_i: サイト S_i の持つ値 ($0 < i < M$)

Output: ΣV_i ($i = 0, 1, ..., M-1$)

begin
  Set セッション数 K to $\lfloor M/2 \rfloor$
  foreach site S_i **do**
    K-1 個の乱数を生成
    各乱数を $Y_{i,j}$ ($2 \leqq j \leqq K$)とする
    $V'_i \leftarrow V_i + \Sigma Y_{i,j}$ ($j = 2, 3, ..., K$)
    foreach $k \in \{1, 2, ..., K\}$ **do**
      Call Session (k, K, M)
    for site S_0 **do**
      do $\Sigma V'_i$ ($i = 0, 1, ..., M-1$) $- \Sigma \Sigma Y_{i,j}$
        ($i = 0, 1, ..., M-1$) ($j = 2, 3, ..., K$)
      return ΣV_i ($i = 0, 1, ..., M-1$)
end

Procedure: Session
(各サイトの値を秘匿しながら集計を行うアルゴリズムの中で呼び出される集計セッション)

Input: (1) k: セッション番号
(2) K: セッション数
(3) M: サイト数

Output: $\Sigma V'_i$ ($i = 0, 1, ..., M-1$) ($k = 1$ のとき)
$\Sigma Y_{i,k}$ ($i = 0, 1, ..., M-1$) ($k > 1$ のとき)

begin
  foreach site S_i **do**
    if $k = 1$ **then** $v_i \leftarrow V'_i$
    else $v_i \leftarrow Y_{i,k}$
    switch the site is in do
      case $\{ S_{j \bmod M-1} \mid j = 0, 2k, ..., 2Kk \}$
        v_i をランダムに 2 つに分ける
        分けられたそれぞれを v'_i, v''_i とする
        ($v'_i + v''_i = v_i$)
        v'_i を $S_{(i+k) \bmod M}$ へ送信
        v''_i を $S_{(i-k+M) \bmod M}$ へ送信
      case $\{ S_{M-1}, S_{j \bmod M-1} \mid j = k, 3k, ..., (2K+1)k \}$
        (1) v_i
        (2) $S_{(i+k) \bmod M}$ から受信した値
        (3) $S_{(i-k+M) \bmod M}$ から受信した値
        以上の合計を S_0 へ送信
  for site S_0 **do**
    Return S_{M-1} と $S_{j \bmod M-1}$ ($j = k, 3k, ..., (2K+1)k$) からそれぞれ受信した値の合計
end

4. 性能評価

本提案アルゴリズムの有効性を確認するため，分析，実装による性能評価を行った．4.1 節では，提案したアルゴリズムの結託耐性，メッセージ数を導出し，先行研究との比較を行った結果を示す．4.2 節では，提案したアルゴリズムを実装し，様々な条件で実行時間を計測した結果を示す．

4.1. 分析による評価

4.1.1. 指標の定義

分析による評価ではその指標として結託耐性とメッセージ数を用いる．結託耐性とは，あるサイトのプライバシーに関わるデータを得るために結託する必要があるサイトの数から 1 を引いた値である．つまり，何個のサイトの結託まで耐えられるかを表す．メッセージ数とは，データマイニングを行う際に，サイトから別のサイトへ渡されるメッセージの数である．これはさらに以下の 2 つのケースに分ける．

ケース 1：全ての通信で発生するメッセージの総数
ケース 2：同時発生するメッセージを 1 つとしてカウントする場合のメッセージ数

4.1.2. 結託耐性の比較

まず，アルゴリズムがどのようにして結託耐性を確保しているかについて述べる．図 1 は，サイト数が 8 の場合，本提案アルゴリズムを用いて値を集計した際の，サイト 4 に関するデータの流れを示した図である．図 1 より，以下のことが分かる．

(1) V'_4 を得るためにはサイト 3, 5 の結託が必要
(2) $Y_{4,2}$ を得るためにはサイト 2, 6 の結託が必要
(3) $Y_{4,3}$ を得るためにはサイト 1, 7 の結託が必要

よって，V_4 を得るためにはサイト 1, 2, 3, 5, 6, 7 及び，$Y_{4,4}$ が送信されるサイト 0 の 7 つのサイト結託が必要となり，結託耐性は 6 となる．他のサイトについても同様の方法で結託耐性が求められる．また，結託耐性はサイト数に応じて増加し，サイト数を M とすると結託耐性は M-2 となる．

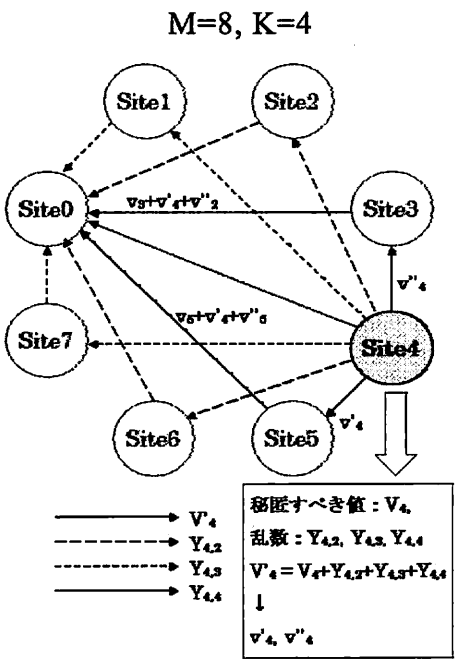

図 1 本提案アルゴリズムのデータの流れ

図 2 は，本提案アルゴリズム，SDDM[3]，SFDM[2]，Crowds[4]の 4 つの方法について結託耐性の比較を行った結果を示したグラフである．SFDM を用いた場合，結託耐性は 1 以下，SDDM ではサイト数が 5 以上の時に常に 3 となる．これらに対し，本提案アルゴリズムを用いた場合の結託耐性はサイト数の増加につれて高くなることが示された．

4.1.3. メッセージ数(ケース 1)の比較

はじめに，本提案アルゴリズムにおけるメッセージ数(ケース 1)の分析について述べる．1 回の集計セッションではM/2+M回の通信が行われる．よって，1つの値の集計では(M/2)*(M/2+M)回の通信が行われることになる．1 回のデータマイニングでは値の集計が 2 回行われ，さらに候補アイテムセットのブロードキャストが 2 回行われるため，全体としては 2*(M/2)*(M/2+M)+2 個のメッセージが発生する．

図 3 は，データマイニングを行う際に発生する総メッセージ数(ケース 1)を比較した結果を示したグラフである．本提案アルゴリズムを用いた場合，SDDM，SFDM に比べて発生するメッセージ数が多くなる．これは，結託耐性の確保のため，サイト数の増加に伴い集計ルートの数(セッション数)も増やしているためである．

4.1.4. メッセージ数(ケース 2)の比較

図 4 は，データマイニングを行う際，同時に発生するメッセージを 1 つとして見た場合の総メッセージ数(ケース 2)を SFDM と比較した結果を示したグラフである．本提案アルゴリズムはサイト数が増えても常に 6 と変わらない値を示した．これは，集計セッションで各サイトのデータ送信を全て並行して行うこと，さらに全ての集計セッションを同時に行うことが可能なためである

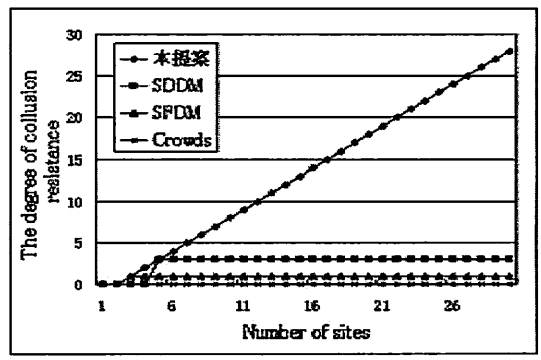

図 2 結託耐性の比較

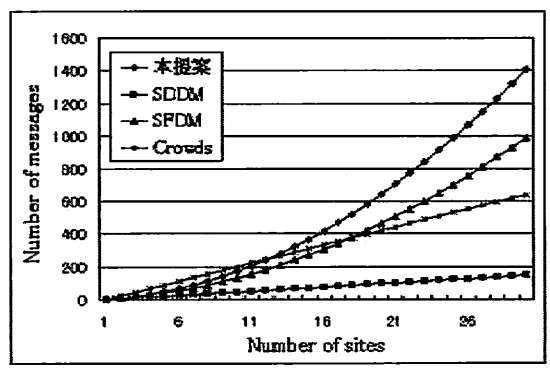

図 3 メッセージ数の比較 (ケース 1)

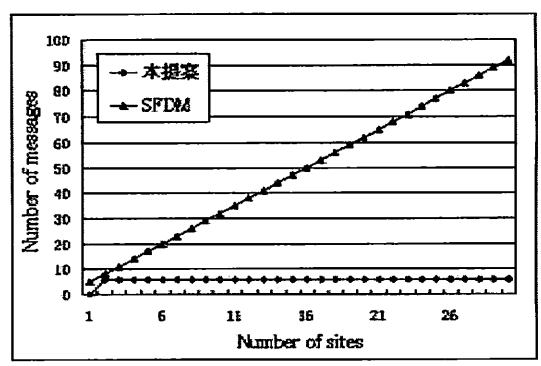

図 4 メッセージ数の比較 (ケース 2)

4.2. 実装による評価

本提案アルゴリズムを実装し，実行時間を比較するための実験を行った．この実験では，本提案アルゴリズムの特性を明確にするため，トランザクション数，最小サポート値をそれぞれ変化させ，実行時間を計測した．

4.2.1. 実装環境

実装の際，プログラムは全て JAVA1.4 を用いて記述した．実装環境は SUN ワークステーション(CPU:Ultra SPARC550MHz, RAM:512 MB)を用い，それぞれのマシンを 100Mbps の LAN 上に接続した．

実験に用いた各トランザクションファイルは IBM が公開しているデータ生成プログラムを用いて作成した．サイト数は 5，各トランザクションの平均の長さは 10，アイテムの種類数は 1000 に設定した．

4.2.2. 実行時間の比較

トランザクション数を変化させた場合の実行時間の比較結果を図 5 に示す．各アルゴリズム名に添えられている括弧内の数字は最小サポート値を示す．図 5 より，本提案アルゴリズムの実行時間は FDM よりやや長く，SDDM より短いことが分かる．

最小サポート値を変化させた場合の実行時間の比較結果を図 6 に示す．各アルゴリズム名に添えられている括弧内の数字はトランザクション数を示す．図 6 より，前述の比較結果と同じく，本提案アルゴリズムの実行時間は FDM よりやや長く，SDDM より短いことが分かる．また，トランザクション数，最小サポート値の増加における実行時間の増加傾向は，先行研究とほぼ変わらないことが分かる．

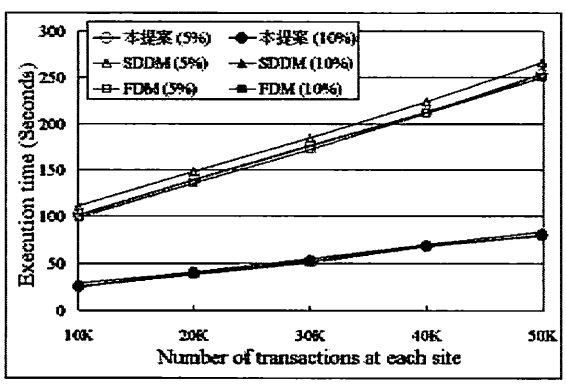

図 5 トランザクション数の変化による実行時間の比較

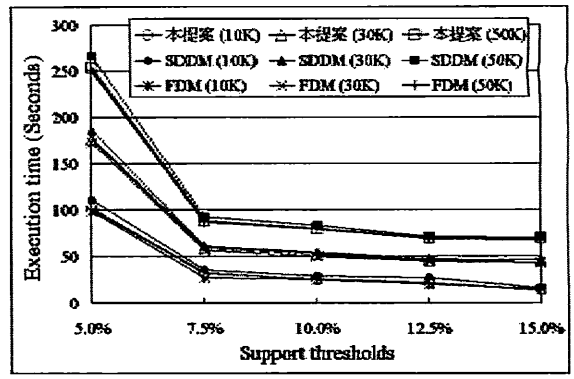

図 6 最小サポート値の変化による実行時間の比較

5. まとめ

本研究では，既存のアルゴリズムよりも結託耐性の高い分散データマイニングアルゴリズムを提案した．さらに，本提案アルゴリズムに対し分析及び実験による評価を行い，その有用性を検証した．その結果，本提案アルゴリズムはサイト数の増加に応じて結託耐性を向上させることが可能になり，先行研究との比較において，既存のアルゴリズムに比べ極めて高い結託耐性を持つことが示された．一方，本提案アルゴリズムは SDDM, SFDM より多くのメッセージが発生する．しかし，実行時間はほぼ変わらず，トランザクション数，最小サポート値が変化しても，実行時間に対する影響は既存のアルゴリズムと変わらないことが示された．以上のことから，本提案アルゴリズムを用いることで，より安全で効率的な分散データマイニングを行うことが可能となった．

文　献

[1] D.W. Cheung, J. Han, V.T. Ng, A.W. Fu, and Y. Fu, "Fast Distributed Algorithm for Mining Association Ruled," in Proc. 1996 International Conference on Parallel and Distributed Information Systems, pp.31-42, Florida, USA, 1996.

[2] M. Kantarcioglu and C. Clifton, "Privacy-preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," IEEE Transactions on Knowledge and Data Engineering, Vol.16, No.9, pp. 1026-1037.

[3] T. Fukazawa, J. Wang, T. Takata, and M. Miyazaki, "An Effective Distributed Privacy-Preserving Data Mining Algorithm," in Proc. Fifth International Conference on Intelligent Data Engineering and Automated Learning, pp.320-325, 2004.

[4] M.K. Reiter and A.D. Rubin, "Crowds: Anonymity for Web Transactions," ACM Transactions on Information System Security, Vol.1, No.1 pp.66-92.

[5] R. Agrawal and R. Srikant, "Fast algorithm for mining association rules," in Proc. the 20th International Conference on Very Large Data Bases, pp.487-499, Santiago, Chile, 1994.