平成 21 年度　修士論文概要

汎用の関係データベースのためのパターンマイニングシステムに関する研究

犬塚研究室　牧野 敏行
学籍番号　No. 20417605

1　はじめに

関係データマイニングとは，複数の関係表で表現されたデータから隠された知識やパターンを発見するプロセスである．この分野は MRDM(Multi-Relational Data Mining) といい，帰納論理プログラミング (Inductive Logic Programming: ILP) で扱われてきた．ILP は述語論理を用いた記述により，豊かな表現力と高い可読性を持っている．

複数の関係表をそのままマイニングすることのできる MRDM 手法は汎用の関係 DB に適している．しかし，ILP の分野で扱われてきた MRDM 手法は，関係 DB とは分離した実装がされてきた．述語論理で記述されたデータを計算機の主記憶上で使用してきたことも問題の 1 つである．

そこで，汎用の関係 DB 中に格納された複数の関係表を対象とした MRDM システムの実装を目標とする．ボトムアップに基本パターンを抽出し，それの組み合わせに限定して高速にパターンを枚挙する手法に Mapix[1] がある．本研究では，RDBMS(関係 DB 管理システム) 上のデータをマイニングできるように Mapix の実装方法を与える．

2　関係データマイニング

簡単な例を用いて MRDM におけるパターンマイニングについて説明する．列車に関するあるデータベース$\mathrm{R}_{\mathsf{train}}$(図 1) について考える．$\mathrm{R}_{\mathsf{train}}$には，列車を登録する関係表 train や，列車がどの貨車を持っているかを表す関係表 has-car，貨車に積まれている荷物の特徴を表す関係表 triangle，circle などの複数の関係表があるとする．関係データマイニングでは，このような複数の関係表から，対象とする関係表 (これを key テーブルと呼ぶ) を選択し，その中のタプルに現れるパターンの中で，与えられた頻度の閾値を満たすパターンを発見していく．そのようなパターンは，頻出パターンと呼ばれる．

train
t_1
t_2
t_3

has-car	
t_1	c_1
t_1	c_2
t_2	c_3
t_3	c_4
t_3	c_5

triangle
c_1
c_3
c_4

circle
c_2
c_5

図 1: key テーブル train を含む 4 つのテーブルを持つデータベース$\mathrm{R}_{\mathsf{train}}$

例えば，この$\mathrm{R}_{\mathsf{train}}$に関して,「列車 A は三角形の荷物を積んだ貨車 B を持つ」というパターンを考えるとすると，このパターンは次の式のように表現される．

$$\mathsf{train}(A) \wedge \mathsf{has\text{-}car}(A, B) \wedge \mathsf{triangle}(B).$$

Mapix では事例集合から，性質という事例に現れる基本パターンを取り出し，性質とその組み合わせに限定して頻出パターンを枚挙する．

2.1　性質

述語には各引数が入力，または出力かを表すモードが定められていて，おのおの +，− で表される．例えば，列車の関係データベース$\mathrm{R}_{\mathsf{train}}$では，各述語モードはhas-car(+, −)，triangle(+)，circle(+)である．

Mapix では，このモードに注目して，述語を二つのクラスに分ける．第一のクラスは全ての引数が入力モードとなっている判定述語 (triangle, circle) である．第二のクラスは入力モードと出力モードをあわせ持つ経路述語 (has-car) である．また，各述語で構成されたリテラルを判定リテラル，経路リテラルと呼ぶ．

このとき，1 つの判定リテラルといくつかの経路リテラルで構成され，引数が注目する対象から判定リテラルの引数まで経路リテラルにより鎖状につながっているリテラル集合を対象についての事実であると考えることができる．このリテラル集合を**性質**と呼び，これは属性の拡張表現とみなすことができる．

2.2　Mapix

以下に Mapix アルゴリズムの流れを示す．

1. 与えられた key テーブルからいくつかの事例を選択し，その事例の関連リテラル集合を生成する
2. 選択した事例に関する関連リテラル集合から性質を抽出する
3. DB 中の性質の頻度を計算し，頻出な性質とその組み合わせに限定して頻出パターンを枚挙する．

事例の関連リテラル集合とは，事例と以下のような関連性をもつ DB 中のリテラルである．

- 事例自身は関連リテラルである．
- DB 中のあるリテラルについてその全ての入力引数が関連リテラルの出力モードの引数の中に現れる場合，そのリテラルも関連リテラルである．

つまり，入力引数の項から出力引数の項へと経由していき，目標の事例へとたどり着くことができる全てのリテラルの集合が関連リテラル集合である．例えば，データベース$\mathrm{R}_{\mathsf{train}}$の事例$\mathsf{train}(t_1)$の関連リテラルは以下のものである．

$$\mathsf{train}(t_1), \mathsf{has\text{-}car}(t_1, c_1), \mathsf{has\text{-}car}(t_1, c_2), \mathsf{triangle}(c_1), \mathsf{circle}(c_2).$$

3　RDBMS へと結合した MRDM 手法の提案

Mapix アルゴリズムの操作を Prolog プログラム側と RDBMS 側とに分割することで実装をおこなう．これによる次の基本手順を提案する．

RDBMS: RDBMS 上の key テーブルから事例を選択し，関連リテラル集合を生成する．

Prolog: 関連リテラル集合を Prolog 側で論理形式に変換し，性質を生成する．

RDBMS: 事例が満たす性質を記録したトランザクションテーブルを生成する．

RDBMS: トランザクションテーブル中の性質の全ての頻出な組み合わせを生成する．

ここで，Prolog 側の操作は Mapix と同様の手法を用いる．ここで，頻出なパターンを枚挙する手法は [2] 等で研究されている．

3.1　関連リテラル集合生成手法の改善

基本手順では，関連リテラル集合を生成するとき，以下のような問題点がある．

- 関係表中の入力モードが 1 つに限定されている．
- 関係表の数が多い場合，結合の回数が増大し，計算量が大きくなってしまう．
- テーブルが保持できる最大の列数を越えてしまう場合，すべての関連リテラルを集められない．
- テーブル中に同じリテラルの情報がいくつも格納されてしまう．
- 一度，集めたリテラル中の項がもう一度現れた場合，同じように集めてしまう．

そのため，以下のアイディアを導入する．

- 関連リテラルのテーブルに関係表の中のリテラルの各項が現れているかどうかの出現検査を行う．
- 同じ情報を持つ関係表を 1 つにまとめる．
- 関係表ごとに関連リテラルをテーブルに格納する．
- 既に集められた関連リテラル中の項の中で処理済みの項と未処理の項を判別し，未処理のものだけから探索していく．

ここでは，特に重要なリテラル中の項の出現検査のアイディアについて説明する．リテラル中の項の出現検査は，入力モードを複数持つ関係表中のリテラルを集める場合に必要となる．複数の入力引数を持つリテラルを関連リテラルとして集める場合，全ての入力引数の項は，既に集められた関連リテラルの出力引数として現れていなければいけないため，それらに対して出現検査がおこなわれる．

4　実装と実験

Prolog と DB を操作するための Open DataBase Connectivity インタフェースを用いて実装をした．

提案手法を 2 つのデータ集合を用いて，元の Mapix との実行時間を比較する．

1 つ目は，英文の文法構造に関するものである．このデータ集合を用いて，トランザクションテーブルを生成するまでの実行時間を計測した．選択する事例数と 5 回の実行の処理時間の平均を表 1 に示す．元の Mapix と比較して短い処理時間を得た．

性質の抽出における事例の選択数 (個)	実行時間 (秒) Mapix	提案手法
100	730	251
1000	7401	1945
3369	25247	6767

表 1: 英文のデータ集合に対する実行時間

2 つ目は Mutagenesis という ILP 問題において利用されてきたデータ集合を用いる．このデータ集合は，突然変異性を持つ有機物質を含む化学物質の構造データであり，複数の入力モードを持つ関係表が含まれている．全事例の 230 個を用いて，先ほどと同様の実験をして動作を確認した．

性質の抽出における事例の選択数 (個)	実行時間 (秒) Mapix	提案手法
230	115	433

表 2: Mutagenesis に対する実行時間

結果としては，Mapix と比較して，提案手法の方が数倍の時間がかかることになったが，これは事例が少ないためオーバヘッドがかかっていると考えられる．

5　まとめ

本研究では，MRDM 手法である Mapix を汎用の関係 DB に接続することによりマイニングをおこなう実装手法を提案した．この提案手法により，関係データマイニングを大規模な DB へと適用するための方向性を示すことができたと思われる．

今後の課題としては，Mapix の機能の一部について実装できていない部分を実装する必要がある．また，マイニングされたパターンの可視化などのインタフェース機能などのシステム全体の課題や，大規模な DB への応用などが課題として残っている．

参考文献

[1] J.Motoyama, S.Urazawa, T.Nakano and N.Inuzuka :“ A Mining Algorithm Using Property Items Extracted from Sampled Examples ”, ILP 2006, S. Muggleton et al. (eds.), LNAI, vol. 4455, Springer, pp.335-350, 2007.

[2] S.Sarawagi, S.Thomas, R.Agrawal :“ Integrating Association Rule Mining with Relational Database Systems: Alternatives and Implications ”, Data Mining and Knowledge Discovery,Vol.4,Nos.2-3, pp. 89-125, Springer, 2004.

発表論文

- N.Inuzuka, T.Makino :“ Implementing Multi-relational Mining with Relational Database Systems ”, KES 2009, LNAI 5712, Springer, pp 672-680, 2009.
- T.Makino, N.Inuzuka :“ Implementing Pattern Mining Using Extended Attribute Expression on Relational DB ”, WKDD 2010, pp.502-505, IEEE Computer Society, 2010.