

極小生成子を用いた負の相関ルール抽出の高速抽出アルゴリズム

A New Fast Mining Algorithm for Negative Association Rules
Based on Minimal Generators

佐生 隼一†
Shunichi Sasho

岩沼 宏治‡
Koji Iwanuma

黒岩 健歩†
Yasuho Kuroiwa

山本 泰生‡
Yoshitaka Yamamoto

1 はじめに

相関ルールの発見はデータマイニングにおける代表的な問題である。相関ルールとはデータベース中で頻繁に共起する事象の関係を記述したものである。\$X\$ と \$Y\$ をアイテム集合とすると、トランザクションデータベース中で \$X\$ が出現するトランザクションの多くに \$Y\$ も出現することを、\$X \Rightarrow Y\$ と記述し、これを正の相関ルールと呼んでいる。これに対して、本研究で考察する負の相関ルールは、ある事象が発生した際に別の事象が生起しない現象を記述するものであり、\$\neg X \Rightarrow Y, X \Rightarrow \neg Y\$ などの形のルールとして記述される。

負の相関ルール抽出問題は近年研究が盛んになった分野 [1, 2, 3] である。正の相関ルールでは表現が困難な共起関係を記述でき、データベースから有益な情報を抽出することを可能にする。ただ、負の相関ルールは非頻出なアイテム集合を扱う必要があり、正の相関ルールと比べて探索空間が格段に大きい。探索の高速化と効果的化は重要な課題であった。これに対して、井出らの先行研究 [4] では高速なトップダウン型アルゴリズムが提案されている。黒岩らの研究 [5] ではルールの統計的評価尺度を併用して負ルールの抽出の高速化を行っている。

本研究では、新しく極小生成子 (minimal generators)[6] を用いた負の相関ルール抽出法を提案する。極小生成子によるアイテム集合の圧縮効果を利用して、負の相関ルール抽出の高速化を行う。頻出アイテム集合の圧縮法は飽和集合がよく知られているが、この飽和集合を圧縮に用いた場合、本来抽出すべき負ルールが抽出できなくなるなどの現象が生じる。極小生成子を用いることにより、完全な負ルールの抽出が可能となる。

本論文の構成は以下の通りである。第 2 章は準備である。第 3 章では先行研究での負ルールの抽出アルゴリズムを示す。第 4 章では、極小生成子を用いた負の相関ルール抽出アルゴリズムを新しく提案する。第 5 章は性能確認のための実証実験の結果と考察を示す。第 6 章はまとめである。

2 準備

\$I = \{a_1, a_2, \dots, a_n\}\$ をアイテムの全体集合とし、トランザクション \$t\$ をアイテム集合 \$t \subseteq I\$ と定める。トランザクションデータベース \$\mathcal{D}\$ をトランザクションの多

重集合とする。\$X\$ をアイテム集合とすると、\$X \subseteq t\$ となる \$\mathcal{D}\$ のトランザクション \$t\$ を \$X\$ の出現と呼び、その多重集合を \$\mathcal{D}(X)\$ と略記する。多重集合 \$A\$ の大きさを \$|A|\$ と表記するとき、\$X\$ の \$\mathcal{D}\$ 中の支持度 \$\text{sup}(X)\$ を \$\text{sup}(X) = \frac{|\mathcal{D}(X)|}{|\mathcal{D}|}\$ と定義する。正の相関ルール (以下、“正ルール” と略記) を \$X \cap Y = \emptyset\$ であるアイテム集合 \$X, Y\$ からなる表現 \$X \Rightarrow Y\$ と定める。\$X\$ と \$Y\$ をそれぞれルールの前件、後件と呼び、\$X \cup Y\$ を台集合 (underlying set) と呼ぶ。正ルールに対する支持度 \$\text{sup}\$ と確信度 \$\text{conf}\$ は以下のように定義する。

$$\text{sup}(X \Rightarrow Y) = \text{sup}(X \cup Y), \quad \text{conf}(X \Rightarrow Y) = \frac{\text{sup}(X \cup Y)}{\text{sup}(X)}$$

最小支持度 \$ms\$ と最小確信度 \$mc\$ とはユーザが与える支持度と確信度に関する閾値である。\$\text{sup}(X) \geq ms\$ を満たす \$X\$ を頻出アイテム集合と呼ぶ。また \$\text{sup}(X \Rightarrow Y) \geq ms\$ と \$\text{conf}(X \Rightarrow Y) \geq mc\$ の両方を満たす \$X \Rightarrow Y\$ を有効 (valid) な正ルールと呼ぶ。

本研究では、負の相関ルール (negative association rule: 以下では“負ルール” と略記) とは、アイテム集合 \$X\$ と \$Y\$ を \$X \cap Y = \emptyset\$ とする時、以下のいずれかの表現のこととする。

$$\neg X \Rightarrow Y \text{ (左否定形)}, \quad Y \Rightarrow \neg X \text{ (右否定形)}$$

上記の \$\neg X\$ はアイテム集合の否定表現であり、負アイテム集合と呼ぶ。以下では \$C_X\$ は、正と負のアイテム集合 \$X\$ または \$\neg X\$ のどちらかを表すものとする。

定義 1 ([1, 2, 3]) 負アイテム集合および負ルールの支持度 \$\text{sup}\$ と確信度 \$\text{conf}\$ を以下のように定める。

$$\begin{aligned} \text{sup}(\neg X) &= 1 - \text{sup}(X) \\ \text{sup}(X \Rightarrow \neg Y) &= \text{sup}(X) - \text{sup}(X \cup Y) \\ \text{sup}(\neg X \Rightarrow Y) &= \text{sup}(Y) - \text{sup}(X \cup Y) \\ \text{conf}(C_X \Rightarrow C_Y) &= \frac{\text{sup}(C_X \Rightarrow C_Y)}{\text{sup}(C_X)} \end{aligned}$$

3 先行研究

先行研究 [1, 2, 3] では、負ルールの抽出は Apriori 流のボトムアップ型アルゴリズムで行われていた。このため、負ルール間の関係性の検査は困難であり、効率的な枝刈りを行うことができなかった。井出ら [4] は接尾木を用いたトップダウン型アルゴリズムを新しく提案し、負ルール間の包含関係を効率的に検査して、探索空間を削減して高速化を達成している。

図 1 に接尾木の例 [7] を示す。アイテム間には適当な順序 \$\prec\$ を仮定し、アイテム集合をアイテムの列としてと

† 山梨大学大学院医学工学総合教育部コンピュータ・メディア工学専攻

‡ 山梨大学大学院総合研究部コンピュータ・メディア工学専攻担当

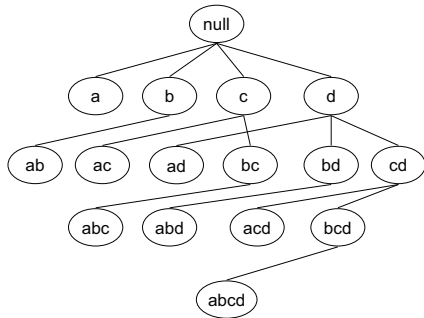


図 1 接尾木

り扱う．図 1 では $a < b < c < d$ を仮定している．各節
点 N_c の親は長さが 1 つだけ短い接尾辞を持つ節点 N_p
である．子 N_c と親 N_p の差分アイテムは \prec 上において
 N_p 中のアイテムより前にあるものに限定される．兄弟
関係にある節点は \prec に基づく辞書順で左から右へ並び、
接尾木上で左優先深さ優先探索を行うと、節点 N を訪
問する時点で N の部分集合は全て訪問が完了している
[7]．これを利用して、先行研究 [4] のトップダウン型アル
ゴリズムでは、包含関係のあるルールの検査を行い、
効率よく探索空間を削減している．

次に先行研究 [5] で負ルールの評価するために導入し
た統計的尺度（以下 関連尺度と呼ぶ）を説明する．[5]
では、正ルール発見問題で用いられる代表的な尺度であ
る lift などを負ルールに適した形に拡張して導入してい
る．lift は、前件と後件の統計的独立性を測る指標であ
り、負ルールに対しては、以下のように定義される．

定義 2 ([5])

$$\text{lift}(C_X \Rightarrow C_Y) = \frac{\sup(C_X \Rightarrow C_Y)}{\sup(C_X) \cdot \sup(C_Y)}$$

先行研究 [5] では以下のような条件を満たすルール
 $C_X \Rightarrow C_Y$ を有効な (valid) 負ルールと定めていた．

定義 3 最小支持度 ms ，最小確信度 mc ，関連尺度の閾
値 mr としたとき、有効な負ルール $C_X \Rightarrow C_Y$ とは以
下の条件を満たすものである．

1. $X \cap Y = \emptyset$
2. $\sup(X) \geq ms$ かつ $\sup(Y) \geq ms$
3. $\sup(X \Rightarrow Y) < ms$
4. $\sup(C_X \Rightarrow C_Y) \geq ms$
5. $\text{conf}(C_X \Rightarrow C_Y) \geq mc$
6. $\text{lift}(C_X \Rightarrow C_Y) \geq mr$

先行研究 [5] では、有効な負ルールを以下の Algo
rithm 1 で抽出を行う．疑似コード中のアイテム集合
 X, Y は接尾木の節点のアイテム集合を示す．2 重 for
文で X, Y の組み合わせについて調べ、有効な負ルール
の抽出を行っている．なお本稿では、高速化のための探
索空間の枝刈り法については省略する．枝刈りの詳細につ
いては [5] を参照して頂きたい．

Algorithm 1 負の相関ルール探索

Input: トランザクションデータベース D ，最小支持度 ms ，最小確
信度 mc ，関連尺度の閾値 mr
Output: 有効な左否定形の負ルールの集合 LN ，有効な右否定形の
負ルールの集合 RN

- 1: D から頻出アイテム集合の集合 $FISS$ を抽出;
- 2: $FISS$ から接尾木を生成;
{ 以下では、接尾木上で深さ優先左優先探索を行ったときの頂点
の並び、即ち頻出アイテム集合の並びを $FIS^1, \dots, FIS^{|FISS|}$
と仮定している }
- 3: $LN := \emptyset$; $RN := \emptyset$
- 4: for $i = 1$ to $|FISS|$ do
- 5: $X := FIS^i$;
- 6: for $j = 1$ to $|FISS|$ do
- 7: $Y := FIS^j$;
- 8: 負ルール $X \Rightarrow \neg Y$ と $\neg Y \Rightarrow X$ の有効性を検査し、有効な
らば LN, RN に適宜追加する;
- 9: end for
- 10: end for
- 11: return (LN, RN) ;

4 極小生成子に基づく負ルール抽出

本研究は、極小生成子を用いて頻出アイテム集合を圧
縮し、アイテム集合の組み合わせを減少させ、有効な負
ルールの抽出の効率化を行う．

頻出アイテム集合の圧縮表現としては以下の飽和アイ
テム集合と閉包集合が良く知られている．極小生成子
(minimal generator) は閉包の対になる表現である．

定義 4 ([6]) アイテム集合 X が飽和しているとは、以
下の条件を満たす X' が存在しない場合を言う．

$$X \subset X', X \neq X' \text{ かつ } \sup(X) = \sup(X')$$

アイテム集合 Y の閉包とは、 $Y \subset X$ かつ $\sup(Y) =$
 $\sup(X)$ を満たす飽和アイテム集合 X のことである． Y
の生成子とは以下の条件を満たす Z を言う．

$$Z \subset Y \text{ かつ } \sup(Z) = \sup(Y)$$

生成子 Z が極小とは、 $Z' \subset Z$ かつ $Z' \neq Z$ となる生成
子 Z' が存在しない場合を言う．

X' を X の閉包または極小生成子とすれば、 X と X'
はデータベース中で必ず同じトランザクションに出現す
る．このとき、 X が出現する相関ルール $\mathcal{R}[X]$ に対し
て、 X を X' に置き換えた相関ルール $\mathcal{R}[X']$ を考えれ
ば、 $\mathcal{R}[X]$ と $\mathcal{R}[X']$ の支持度、確信度および関連尺度は
全く同じになる．この 2 つのルール両方を生成すること
は冗長と考えられ、どちらかで代表させることが適切で
ある．

以上の考察に基づき、頻出アイテム集合を閉包もしく
は極小生成子で圧縮することを考える．例 1 に示すよう
に、飽和集合を用いると本来抽出すべき妥当なルールが
抽出できなくなる場合がある．極小生成子を用いること
により妥当な負ルール全てが抽出可能になる．

例 1 表 1 のトランザクションデータベース D を考え、
最小支持度 $ms = 0.4$ ，最小確信度 $mc = 0.4$ とする．本
例では、議論の簡単化のために、関連尺度は考慮しない．
また、アイテム集合はその要素の列で表記し、出現頻度

表1 トランザクションデータベース D

TID	アイテム集合
1	ABC
2	AB
3	AB
4	BC
5	BC

を適宜付記する．例えば，出現頻度3のアイテム集合 $\{A, B\}$ を $AB:3$ のように表記する．

データベース D の頻出アイテム集合は以下の通りである．

$$A:3, B:5, C:3, AB:3, BC:3$$

この中で，頻出飽和アイテム集合は B, AB, BC の3つである．このとき以下の負ルール \mathcal{R} を考える．

$$\mathcal{R} = (AB \Rightarrow \neg C)$$

$\text{sup}(\mathcal{R}) = 0.4 \geq ms$, $\text{conf}(\mathcal{R}) = \frac{2}{3} \geq mc$ であり，後件の否定を外した正ルールに対しても $\text{sup}(AB \Rightarrow C) = \frac{1}{5} < ms$ なので， \mathcal{R} は妥当な負ルールである．一方で頻出アイテム集合 C の閉包は BC であることから， \mathcal{R} の飽和集合による表現は

$$AB \Rightarrow \neg BC$$

となる．しかしこれは定義3の(1)に示した前件と後件の独立性条件に違反し，妥当なルールにはならない．これに対して， D 上の頻出アイテム集合に対する極小生成子は A, B, C なので，極小生成子を用いた \mathcal{R} の表現は

$$A \Rightarrow \neg C$$

となる．独立性の条件を満足するので妥当な負ルールとして生成できる．

極小生成子は，圧縮に伴う妥当なルールの表現問題を解決できる．更に，接尾木の深さ優先探索による負ルール生成手法の効率化にも有効である．接尾木の深さ優先探索では，小さい頻出アイテム集合から探索を行っていく．アイテム集合 X の極小生成子を X' とするとき，接尾木探索では X' を必ず先に探索するので， X についてスキップすることが可能となり，高速化できる．

極小生成子を用いた負ルール探索法の疑似コードを Algorithm 2 と Algorithm 3 に示す．Algorithm 2 は Algorithm 1[5] に極小生成子の検査を加えたものである．前件 X と後件 Y の両方に対して極小生成子の判定を追加している．前件 X についてスキップできる場合は，後件 Y に関する内側の for 文の計算を全てスキップできるため，実行時間を大きく減少させることができると思われる．

Algorithm 3 では，極小生成子の検査を行っている．アイテム集合 X の要素数 $|X|$ が1であれば， X は無条件に極小生成子となる． $|X| \geq 2$ であれば，要素数が1つ小さい X の部分集合 $Z_1, \dots, Z_{|X|}$ の支持度を検査し，全ての支持度が X よりも大きければ， X は極小生成子であることに着目して判定する．

Algorithm 2 極小生成子を用いた負の相関ルール探索

Input: トランザクションデータベース D , 最小支持度 ms , 最小確信度 mc , 関連尺度閾値 mr

Output: 有効な左否定形の負ルールの集合 LN , 有効な右否定形の負ルールの集合 RN

```

1:  $D$  から頻出アイテム集合の集合  $FISS$  を抽出;
2:  $FISS$  から接尾木を生成;
   { 以下では，接尾木上で深さ優先左優先探索を行ったときの頂点の並び，即ち頻出アイテム集合の並びを  $FIS^1, \dots, FIS^{|FISS|}$  と仮定している }
3:  $LN := \emptyset$ ;  $RN := \emptyset$ 
4: for  $i = 1$  to  $|FISS|$  do
5:    $X := FIS^i$ ;
   { 以下で  $X$  が極小生成子であるか否かを検査 }
6:   if  $MGCheck(X) = \text{true}$  then
7:     for  $j = 1$  to  $|FISS|$  do
8:        $Y := FIS^j$ ;
       { 以下で  $Y$  が極小生成子であるか否かを検査 }
9:       if  $MGCheck(Y) = \text{true}$  then
10:        負ルール  $X \Rightarrow \neg Y$  と  $\neg Y \Rightarrow X$  の有効性を検査し，有効ならば  $LN, RN$  に適宜追加する;
11:      end if
12:    end for
13:   end if
14: end for
15: return  $(LN, RN)$ 

```

Algorithm 3 $MGCheck(X)$

Input: 頻出アイテム集合 X

Output: X が極小生成子であれば true, さもなくば false.

```

1: if  $|X| = 1$  then
2:   return true;
3: else
4:   {  $|X| \geq 2$  場合 要素数  $|X|-1$  の部分集合  $Z_1, \dots, Z_{|X|}$  を検査 }
5:   for  $i = 1$  to  $|X|$  do
6:     if  $\text{sup}(X) = \text{sup}(Z_i)$  then
7:       return false;
       {  $X$  が極小生成子ではないので false を返す }
8:     end if
9:   end for
10:  return true;
11: end if

```

5 実験結果及び考察

実験には，Frequent Itemset Mining Dataset Repository[8] から5種のデータセットを使用した．各データセットの詳細を表2に示す．mushroom, connect, retail, kosarak は実データ，T10I4D100K はランダムデータである．また，mushroom, connect は稠密 (dense) なデータセットであり，retail, kosarak, T10I4D100K は疎 (sparse) なデータセットである． $\#(\text{item})$ はデータセット中に含まれるアイテムの種類数を示し， $\#(\text{trans.})$ はトランザクションの総数， $\text{ave}(\text{item})$ は各トランザクション中に出現するアイテムの平均数である． $\#(\text{FIS})$ はそれぞれ頻出アイテム集合の数である．検査対とは，探索を行った頻出アイテム集合の対 $\langle X, Y \rangle$ の総数である．

最小確信度 $mc = 0.4$, 追加した関連尺度を lift を用いて， $mr = 1.0$ に固定し，最小支持度 ms の値を変化

表 3 実験結果

データセット	ms	#(FIS)	手法	検査対	探索時間 (sec)	抽出ルール数
mushroom	0.35	1189	先行研究	339,992	1.31	10,721
			提案手法	50,353	0.59	3,959
	0.4	565	先行研究	94,032	0.48	4,720
			提案手法	16,300	0.28	1,646
	0.45	329	先行研究	30,436	0.22	1,651
			提案手法	5,612	0.13	473
connect	0.95	2,205	先行研究	277,966	0.41	17,992
			提案手法	101,128	0.40	5,752
	0.96	1,031	先行研究	89,115	0.25	10,203
			提案手法	40,354	0.22	3,940
	0.97	487	先行研究	24,699	0.16	4,902
			提案手法	14,200	0.16	2,716
retail	0.002	2,715	先行研究	2,872,876	8.6	1,815
			提案手法	2,872,876	9.2	1,825
	0.003	1,409	先行研究	831,152	4.2	1,823
			提案手法	831,152	4.2	1,823
	0.004	835	先行研究	319,823	2.3	1,610
			提案手法	319,823	2.3	1,610
kosarak	0.04	42	先行研究	1008	6.59	482
			提案手法	1008	6.72	482
	0.05	33	先行研究	563	6.44	258
			提案手法	563	6.20	258
	0.06	24	先行研究	352	6.14	124
			提案手法	352	5.90	124
T10I4D100K	0.01	385	先行研究	3,540	0.33	693
			提案手法	3,540	0.33	693
	0.02	155	先行研究	23,870	0.94	593
			提案手法	23,870	0.90	593
	0.03	60	先行研究	3,540	0.33	433
			提案手法	3,540	0.33	433

表 2 実験に使用したデータベース

データベース	#(item)	#(trans.)	ave(item)
mushroom	120	8,124	23.0
connect	130	67,557	43.0
T10I4D100K	870	100,000	10.1
retail	16,470	88,162	10.3
kosarak	41,270	990,002	7.1

させて負ルールを抽出した実験結果を表 3 に示す。

実験データより、密なデータセットにのみ検査対及び抽出ルール数に減少が見られる。mushroom, connect については約 60% にルール数が圧縮されている。疎なデータセットでは検査対等が全く減少しない。これは、全ての頻出アイテム集合が極小生成子となり、スキップが起きなかったためと考えられる。但し、先行研究と比較すると実行時間に殆ど差がなく、極小生成子の検査のオーバーヘッドは非常に小さいことが分かる。これらから、負ルール抽出法の高速化法として価値があると考えられる。

6 まとめ

本研究では、先行研究 [2] で提案された負の相関ルール抽出アルゴリズムに極小生成子を導入し、負の相関ルール抽出の高速化を行う手法を提案した。極小生成子を用いることで、密なデータセットに対しては有効な効果を示すことができた。極小生成子とは有効な概念であることから、今後、更なる工夫を行い、より高速なアルゴリズムの開発を行う予定である。

謝辞: 本研究の一部は ISPS 科学研究費補助金

(25330256) の援助を受けている。

参考文献

- [1] C. Cornelis, P. Yan, X. Zhang, and G. Chen: Mining Positive and Negative Association Rules from Large Databases. *Proc. CIS 2006*. LNCS, Vol.4456, pp.613–618, 2006.
- [2] H. Wang, X. Zhang and G. Chen: Mining a Complete Set of Both Positive and Negative Association Rules from Large Databases. *Proc. the PAKDD '08*, pp.777–784, 2008.
- [3] X. Wu, C. Zhang, and S. Zhang: Efficient Mining of Both Positive and Negative Association Rules. *ACM Trans. on Information Systems*, Vol.22(3), pp.381–405, 2004.
- [4] 井出典子, 岩沼宏治, 山本泰生: 負の相関ルールを抽出する高速トップダウン型アルゴリズム, 人工知能学会論文誌 29 巻 4 号, pp. 406-415 (2014).
- [5] 黒岩健歩, 岩沼宏治, 山本泰生: 関連尺度に基づいた負の相関ルール抽出手法の高機能化, 第 28 回人工知能学会全国大会, 3J3-3in, (2014).
- [6] M. J. Zaki: Mining Non-Redundant Association Rules. *Data Mining and Knowledge Discovery*, Vol.9, pp.223-248 (2004)
- [7] 亀谷由隆, 佐藤泰介: 最小サポート上昇法に基づく上位 k 関連パターン発見. データ指向構成マイニングとシミュレーション研究会 SIG-DOCMAS B101-4, pp.(2-24)–(2-32), 2011.
- [8] Frequent Itemset Mining Dataset Repository, <<http://fimi.ua.ac.be/>>(2015-6-23).