VNU Journal of Mathematics – Physics, Vol. 29, No. 2 (2013) 48-54

Closure mappings and the problem of determining maximal frequent itemsets in data mining

Bui Duc Minh*

Deparment of IT, Ho Chi Minh City College of Transport, Vietnam

Received 10 January 2013
Revised 16 March 2013; Accepted 25 March 2013

Abstract: In data mining, association rules are considered as a fundamental problem. Process of association rules can be run in two stages. The first stage is to find all the frequent itemsets, and the second stage is to generate association rules. However, with a large database, the number of itemsets will be very large and thus the problem of finding association rules is not feasible. In this paper, the author uses he notation of closure mappings and lattice theory as a mathematical approach to show the applicability of these tools to the data mining. In particular, a method of determining maximal itemsets with the purpose of minimal scanning times of database is presented in the paper.

Keywords: Closure mapping, Intersection lattice, maximal frequent itemset, coatom.

1. Basic concepts

Closure mapping is an operator determining correlation between subsets of a given limited set. The mapping is satisfied reflexibility, monotonicity, and idempotence proerties. Researching in general about closure mappings and intersection lattices allows expanding the applying some mathematical tools to develop and apply some results in many fields, especially in data mining.

The aim of the paper is presentation of using closure mapping and intersection lattice theory in data mining. The first result of the paper is affirmative clause that the frequent itemsets family in a transaction database forms an intersection lattice [2]. From that, we apply properties of intersection lattice to determine maximal frequent itemsets of a frequent itemsets family. The paper proposes a method to determine maximal frequent itemsets in process of generating association rules with minimum of itemsets, improve computational performance, especially in large data.

There are four sections in this paper. The first section presents basis concepts of closure mapping and intersection lattice theory, the common concepts and properties in data mining is presented in the

* Tel.: 84- 903687898
E-mail: buiducminh@gmail.com

second section. The coatom algorithm and related algorithms for detemining maximal frequent itemsets are presented in the third section, and the last section is conclusion.

Definition 1.1 [1]

Given a limited set U, *SubSet*(U) is a set containing all subsets of U. Mapping f: SubSet(U) $\rightarrow$ SubSet(U) is called *closure* on the set U if

$\forall X, Y \subseteq U$:

(i) Reflexibility: $f(X) \supseteq X$,

(ii) Monotonicity: if $X \subseteq Y$ then $f(X) \subseteq f(Y)$,

(iii) Idempotence : $f(f(X)) = f(X)$.

Definition 1.2 [1]

Let f be a given closure mapping on limited set U. Subset $X \subseteq U$ is called a *fixed point* or closed subset of f if $f(X) = X$.

The set of all fixed points of a closure mapping f on U is denoted by $Fix(f)$. Due to $f(U)=U$, thus $Fix(f)$ always contains U as the biggest element. Besides, based on the idempotence of closure mappings, we can represent $Fix(f)$ as: $Fix(f) = \{ f(X) \mid X \subseteq U \}$.

If $X, Y \in Fix(f)$. Then $X \cap Y \subseteq X$ and $X \cap Y \subseteq Y$. By monotonicity of f, we have $f(X \cap Y) \subseteq X$ and $f(X \cap Y) \subseteq Y$. This implies $f(X \cap Y) \subseteq X \cap Y$. *Conversely, by reflexibility of f, we have* $X \cap Y \subseteq X \subseteq f(X)$ and $X \cap Y \subseteq Y \subseteq f(Y)$. This implies $X \cap Y \subseteq f(X \cap Y)$. Combining $f(X \cap Y) \subseteq X \cap Y$ *and* $X \cap Y \subseteq f(X \cap Y)$ we have $f(X \cap Y) = X \cap Y$. That is, $X \cap Y$ is a closed set, $X \cap Y \in Fix(f)$. We say that, $Fix(f)$ *is closed on the set−intersection operation.*

Definition 1.3 [1]

Let G be a family of a given limited set U. Suppose that G is closed on the set−intersection operation, thus the intersection of every sub-family in G returns a subset in G,

$$G \subseteq SubSet(U): (\forall H \subseteq G \Rightarrow \bigcap_{X \in H} X \in G)$$

G is called an intersection lattice in a limited set U.

Let G be an intersection lattice in a limited set U. Then G contains an unique sub-family S such that every element of G is represented by intersection of elements in S. It is known that S is the smallest subset of G satisfied property:

$$G = \{ X_1 \cap \ldots \cap X_k \mid k \geq 0, X_1, \ldots, X_k \in S \}$$

S is called a generator of lattice G and denoted as $Gen(G)$, $S = Gen(G)$

Following convention, intersection of empty family of subsets is U, so every intersection lattice contains U and U doesn't belong to $Gen(G)$.

From now, we suppose that a limited subset $U \neq \emptyset$ is always given.

In intersection lattice theory of closure mapping, the generator plays a basis role, the following theorem shows how to represent a generator set with many meanings.

Theorem 1.1 [1]

Let G be a intersection lattice in a limited set U. Then four following sets are the same:

(i) $Gen(G)$

(ii) $\{ V\in G \mid V\neq U, (\forall X,Y\in G, X\neq V, Y\neq V) \Rightarrow X\cap Y\neq V \}$

(iii) $\{ V\in G \mid V\neq U, (V=X_1\cap\ldots\cap X_k; X_1,\ldots,X_k\in G, k\geq 1) \Rightarrow (\exists i, 1\leq i\leq k: V = X_i) \}$

(iv) $\{ V\in G \mid V\subset \bigcap_{\substack{X\in G \\ V\subset X}} X \}$

Definition 1.4 [1]

Let $(M, \leq)$ be a limited set with partial order. Element m in M is called *maximal* if $m \leq x$ and $x\in M$, we always have $m=x$. Let $MAX(M)$ be the set of maximal elements of M. It is known that, $\forall x\in M$,$\exists m\in MAX(M)$: $x \leq m$.

Proposition 1.1 [1]

Let $(M, \leq)$ be a limited set with partial order and $P \subseteq Q \subseteq M$. Then if $x\in MAX(Q)$ and $x\in P$ then $x\in MAX(P)$.

Definition 1.5 [1]

Let G be an intersection lattice in U. It is denoted by $Coatom(G) = MAX(G \setminus \{U\})$ and elements in $Coatom(G)$ is called *Co-atom* of G.

Lemma 1.1 [1]

For every intersection lattice G in a limited set U, we have: $MAX(Gen(G)) = MAX(G\setminus\{U\})$

2. Problem of Frequent itemsets mining

Definition 2.1 [4,5]

A transaction database is a pair of $\alpha = (T, I)$ where $I = \{x_1, x_2, \ldots, x_n\}$ is a set of items and $T = \{t_1, t_2, \ldots, t_m\}$ is the set of transactions in α. In this paper, each transaction $t\in T$ is presented by a binary vector, if the i^{th} value is 1, then the item x_i appears in t.

Definition 2.2 [4,5]

Given a transaction database α and itemset $X \subseteq I$. The *support of* X in α is the number of transactions in α containing X, denoted $\sigma(X)$.

Definition 2.3 [4,5]

The set $X \subseteq I$ is *frequent* if $\sigma(X) \geq minsup$, where minsup is a frequent threshold which is determined by the user.

Property 2.1 [4,5]

Let X be a frequent itemset. Then all non–empty subsets of X are frequent.

Proposition 2.1 [2]

Let P be a family of all frequent itemsets in $\alpha = (T, I)$. Then P is an intersection lattice.

Proof

Suppose $X, Y \in P$, $Z = X \cap Y$. We have $Z \subseteq X$, so $\sigma(Z) \geq \sigma(X) \geq minsup$. Thus, $Z \in P$. Following the definition 1.3, P is a intersection lattice.

Definition 2.4 [4,5]

Given a transaction database $\alpha = (T, I)$ and itemset $X \subseteq I$. We say that X is the *maximal frequent* itemsets if X is frequent itemset and X is not pure subset of any frequent itemset at all. Notation *MFI* is family of maximal frequent itemset of α.

Property 2.2

For any frequent itemset, there exists a maximal frequent itemset containing it.

Proof

Let call family of frequent itemsets and maximal frequent itemsets be P and *MFI*. Suppose that $X \in P$,and $X \notin MFI$. If not exist set $Y \in MFI$ such that $X \subseteq Y$, following definition 2.4 then X is maximal frequent itemset, or $X \in MFI$. This is against supposition. So each frequent itemset always exists a maximal frequent itemset containing it.

Remark 2.1

From property 2.2, we see that in process of generating association rules by parent-child relationship, instead of managing all gained frequent itemsets, we only determine and manage maximal frequent itemsets to be sure that generating of association rules is sufficient.

3. Algorithm of finding maximal frequent itemsets

To determine family of frequent itemsets, in previous papers, authors proposed and improved better than many algorithms such as Apriori, Eclat, Declat,… to reduce time. The purpose of this paper is presenting the ability of using closure mapping and intersection lattice in data mining, for simplicity, we use Apriori algorithm to determine family of frequent itemsets in Coatom Algorithm to find maximal frequent itemsets.

3.1 Coatom Algorithm

From given transaction databas, we use Apriori algorithm [3] to determine family of frequent itemsets. Then, Coatom algorithm will build a directed graph H to determine family of maximal frequent itemsets.

```
Algorithm Coatom
Input:   - α = (T,I), minsup
Output:  - MFI
Method
```

```
   1. P = Apriori(T,I,minsup)
   2. Build a directed graph H, each vertex is an element of P, edge
       X → Y if X covers Y, it means that Y ⊂ X and not exist
       element Z ∈ P satisfied Y ⊂ Z ⊂ X
   3. Return MFI = { X ∈ P | I → X }
End Coatom

Algorithm Apriori
Input:   - α = (T,I), minsup
Output:  - Family of frequent itemsets P
Method
   L1 = { j ∈ I: σ(j) ≥ minSup}
   For (k = 2; Lk-1 ≠ ∅; k++) do
      Ck = Apriori_gen(Lk-1)
      For each t ∈ T do
         For each ck ∈ Ck do
            If ck ⊆ t then ck.count++
      Lk = {ck ∈ Ck | ck.count ≥ minSup}
   Return P = ∪k Lk
End Apriori

Algorithm  Apriori_gen(Lk-1)
Method
   Ck = ∅
   For each l1 ∈ Lk-1 do
      For each l2 ∈ Lk-1 do
         If(l1[1]=l2[1])∧(l1[2]=l2[2])∧...∧(l1[k-1]<l2[k-1])then
            c = l1 ∪ l2
            If NOT Has_infrequent_subset(c, Lk-1)then
               Add c into Ck
   Return Ck
End Apriori_Gen

Algorithm Has_Infrequent_Subset(c,Lk-1)
Method
   for each (k-1)-itemset s ⊂ c do
      if s ∉ Lk-1 then
         Return True
   Return False
End Has_Infrequent_Subset
```

3.2 Example

Given transaction database $\alpha = (T, I)$ where $T = \{1,2,3,4,5,6\}$, $I = \{A,C,D,T,W\}$ in following table:

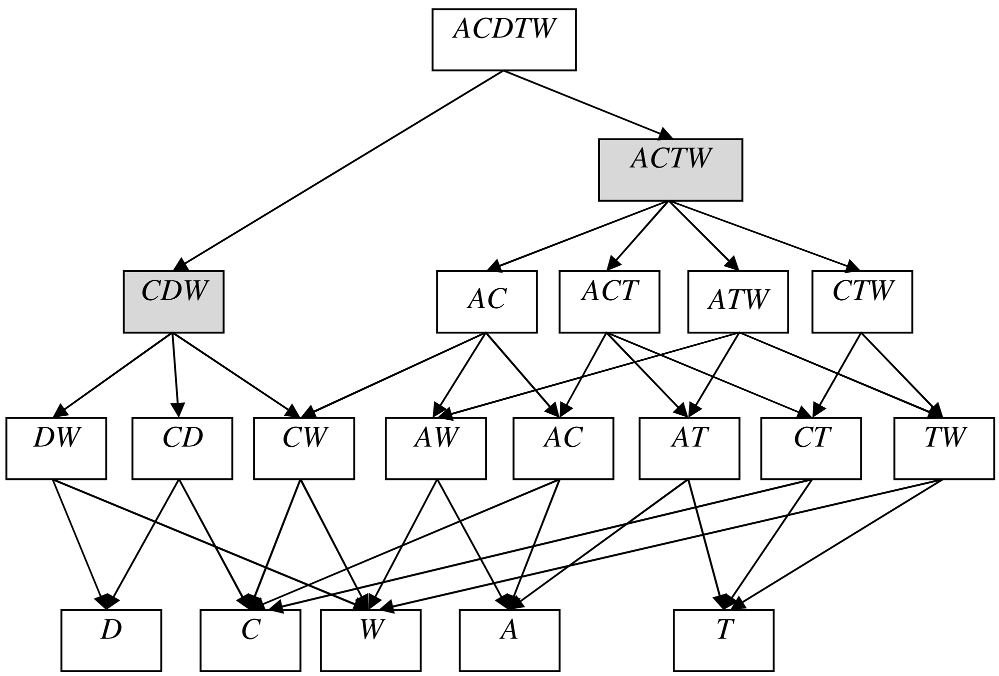
Table 3.1. *Database* α =(*T*, *I*)

Transaction	Item
1	*A, C, T, W*
2	*C, D, W*
3	*A, C, T, W*
4	*A, C, D, W*
5	*A, C, D, T, W*
6	*C, D, T*

With support threshold *minsup*=3. By *Apriori* algorithm, we have list of frequent itemsets such as:

P = {*A, C, D, T, W, AC, AT, AW, CD, CT, CW, DW, TW, ACT, ACW, ATW, CDW, CTW, ACTW*}.

From family of frequent itemsets *P*, we bild a directed graph *H,* where each vertex is an element of *P*, edge $X \rightarrow Y$ if *X* covers *Y* by *Coatom* algorithm:



Figue 3.1. Lattice of frequent itemsets

In graph above and Coatom algorithm, we determine family of maximal frequent itemsets *MFI* = {*CDW, ACTW*}.

4. Conclusion

This paper presents an application of closure mapping and intersection lattice theory in determining maximal frequent itemsets of lattice by Coatom algorithm. From family of maximal frequent itemsets, it is very easy to generate association rules instead of managing too much frequent itemsets, especially in large databases.

References

[1] Nguyen Xuan Huy, Logic Dependencies in Database, Institute of Information Technology, Statistical Publishing House (2006).

[2] Nguyen Xuan Huy, Le Quoc Hai, Nguyen Gia Nhu, Cao Tung Anh, Bui Duc Minh, The theory of Lattice and application in hiding sensitive frequent itemsets, The 15th National Symposium of selected ICT Problems,, BienHoa, VietNam, (2009), pp 161-170.

[3] Rakesh Agrawal, Ramarkrishnan Srikant, Fast Algorithms for Mining Association Rules, Proceedings of VLDB'94, Santiago, chile, 487-499, 1994

[4] Mohammed J. Zaki, Mitsunori Ogihara, Theoretical Foundations of Associations Rules, Proceeding of 3rd SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Seattle, WA, USA 1998.

[5] Mhammed J Zki, Mining non-redundant Associations Rules, Data Mining and Knowledge Discovery 9 (3), 223-248, 2004.

[6] Mohammed J. Zaki and Ching-Jui Hsiao charm: Efficient Algorithm for Mining Closed Itemsets and Their Lattice Structůe. IEEE Transactions On Knowledge And Data Engineering Vol 17 No 4 April 2005.

[7] Karam Gouda, Mohammed J.Zaki, Genmax: An Efficient Algorithm For Mining Maximal Frequent Itemsets, Data Mining and Knowledge Discovery, 11, 1-20, 2005 © 2005 Springer Science + Business Media, Inc. Manufactured in The Netherlands

[8] S.S.Mantha, Madhuri Rao, Ashwini Anilmane, Anil S. Mane, Mining Maximal Frequent Item Sets, International Journal of Computer Applications (0975-8887), Vol 10-No.3, November 2010.

[9] M.Rajalakshmi,T.Purusothaman, R.Nedunchezhian, Maximal Frequent Itemset Generation Using Segmentation Approach, International Journal of Database Management Systems (IJDMS), Vol.3, No.3, August 2011.