Association Mining and Formal Concept Analysis [*]

Jitender S. Deogun[†] Vijay V. Raghavan[‡] Hayri Sever[§]

Abstract

In this paper, we develop a connection between association queries and formal concept analysis. An association query discovers dependencies among values of an attribute grouped by other, non-primary attributes in a given relation. Formal concept analysis deals with formal mathematical tools and techniques to develop and analyze relationship between concepts and to develop concept structures. We show that dependencies found by an association query can be derived from a concept structure.

Keywords- Association queries, formal concept analysis, dependency relations, concept structures.

1 Introduction

An association query discovers dependencies among values of an attribute grouped by some other attributes in a given relation. A specific case of discovering associations concerns with a concrete problem that focuses on the analysis of market-basket-data (or, simply, basket relation) and in the end the solution of market-basket problem helps a retail store to learn about its customers' purchasing trends. Consider the following example.

Example: The following set defines many-to-many association between *products* and *sale transactions* and each entity in this relationship set is described by *quantity*. The attribute names T,P, and Q respectively represent transaction number, product id and quantity. The transaction table can be represented by a relation $r(T, P, Q)$ as follows:

```
{<100,1,3>, <100,2,1>, <100,4,5>,
 <200,1,1>, <200,4,1>, <200,3,1>,
 <300,1,1>, <300,2,1>, <300,4,4>,
 <300,5,2>, <400,2,3>, <400,5,3>,
 <400,6,2>, <400,7,2>, <400,8,7>,
 <500,1,6>, <500,2,3>, <500,7,4>,
 <600,2,1>, <600,4,1>, <600,7,2>,
 <600,1,3>},
```

where r is a relation of arity three and T and P are key attributes. This relation r might be a result of joining certain weak-entities to owner entities, e.g., the set of products with respect to a distinct transaction number in a retail store point-of-sale data.

Assume that we are interested in discovering which products are sold together. This can be done by grouping the values of P with respect to T in r, i.e., SELECT P' FROM r GROUP BY T AS $r_1(P')$. For the sake of simplicity, we assume that SQL allows multi-valued attributes in a relation [1]. That is, $r_1(P') =$

```
{<{1,2,4}>,    <{1,4,3}>,<{1,2,4,5}>,
 <{2,5,6,7,8}>,<{1,2,7}>,<{1,2,4,7}>}.
```

This type of set is called basket relation as shown in Table 1.

Stated another way, let $\Im(P)$ denote all possible subsets of the power set of the domain P. The objective is to find interesting associations $R \subseteq P' \times P'$, where $P' \in \Im(P)$, provided that minimum confidence, say c, and support, say s, are satisfied where $c = Pr(y|x)$ and $s = Pr(x \cup y)$ given a xRy for $x, y \in \Im(P)$.

Let us illustrate the computation of confidence and support measures for xRy, where $x = \{1\}$ and $y = \{2\}$, in the nested relation $Basket(T, R1(P, Q))$. Assume that $Pr(x)$ shows the probability that all the products in x are present in a Basket entity. Then, $c(xRy) = Pr(2|1) = 4/5 = 0.8$ and $s(xRy) = Pr(1 \cup 2) = 4/6 = 0.66$ [2].

This problem can be converted into a Boolean relation whose tuples include only ones and zeroes [2]. Let the size of P be n, that is, n is equal to the number of values in the domain of P denoted by $dom(P)$. Let $f : X \in P' \Rightarrow < B_1, B_2, ..., B_n >$ be a function such that for a given subset X of P it returns a tuple of arity n in which B_i is one if the ordinary order of $x \in X$ is i in $dom(P)$; otherwise

[*] This research was supported in part by the Army Research Office, Grant No. DAAH04-96-1-0325, under DEPSCoR program of Advanced Research Projects Agency, Department of Defense and by the U.S. Department of Energy, Grant No. DE-FG02-97ER1220.

[†] deogun@cse.unl.edu, The Department of Computer Science, University of Nebraska, Lincoln, NE 68588, USA

[‡] CONTACT PERSON raghavan@cacs.usl.edu, The Center for Advanced Computer Studies, University of SW Louisiana,Lafayette, LA 70504, USA

[§] sever@eti.cc.hun.edu.tr, The Department of Computer Science, Hacettepe University, 06532 Beytepe, Ankara, TR

[1] We refer the reader to [1] for discussion of SQL extensions so that some important portion of data mining queries like association and sequential queries are supported.

[2] Strictly speaking, an event in our context corresponds to a subset of elements in $\Im(P)$, but we choose to remove set notation if the event is a single set, e.g., we mean $Pr(\{1\}|\{2\})$ by $Pr(1|2)$.

Transaction No.	Product id	Quantity
100	1	3
	2	1
	4	5
200	1	1
	4	1
	3	1
300	1	1
	2	1
	4	4
	5	2
400	2	3
	5	3
	6	2
	7	2
	8	7
500	1	6
	2	3
	7	4
600	1	3
	2	1
	4	1
	7	2

Table 1: Basket relation grouped by transactions

B_i is assigned zero. Then we obtain $r_2(B_1, B_2, ..., B_n)$ from $r_1(P')$ using the tuple relation expression $\{t_2 \mid \exists t_1 \in r_1(t_2 = f(t_1))\}$.

If we use our running example in Table 1 then $r_2(B_1, B_2, B_3, B_4, B_5, B_6, B_7, B_8) =$

```
{<1,1,0,1,0,0,0,0>,<1,0,1,1,0,0,0,0>,
 <1,1,0,1,1,0,0,0>,<0,1,0,0,1,1,1,1>,
 <1,1,0,0,0,0,1,0>,<1,1,0,1,0,0,1,0>}.
```

In other words, the association problem can be transformed to the problem of discovering association between 1's in some relation.

This type of association is called Boolean association problem [2]. Typically an association query strives for discovering a dependency between two subsets of values of an attribute with respect to externally defined parameters like minimum support and confidence [3]. In next section, we argue that the framework for formal concepts [4, 5] can be used as a natural basis for the analysis of association queries.

Transaction No.	Product Id							
	1	2	3	4	5	6	7	8
100	×	×		×				
200	×		×	×				
300	×	×		×	×			
400		×			×	×	×	×
500	×	×					×	
600	×	×		×			×	

Table 2: Context for Basket relation

2 Use of Concept Lattice for Associations

A *context* is defined as a triple (G, M, I), where G and M are sets and $I \subseteq G \times M$. The elements of G are regarded as objects and the elements of M as features or attributes that the objects might have. For the object g and the attribute m, $(g, m) \in I$ or more commonly, gIm implies that 'the object g possesses the attribute m'. An example of a context (G, M, I) is shown in in Table 2 that is transformed from the Basket relation in Table 1. For this example $G = \{100, 200, 300, \ldots, 600\}$ is a set of six retail transactions, and $M = \{1, 2, 3, \ldots, 8\}$ is a set of eight meta attributes containing product-id as their own name. The formal concept analysis pioneered by Wille is based on the premise that a concept is a pair, one element of which is called its extent and the other its intent [4]. The *extent* of a concept is a set $A \subseteq G$ of objects, and the *intent* is a set $B \subseteq M$ of features shared by objects in A. Let $A \subseteq G$ and $B \subseteq M$ and let

$$\tau(A) = \{m \in M \mid \forall g \in A(gIm)\},$$

$$\varepsilon(B) = \{g \in G \mid \forall m \in B(gIm)\}.$$

Because of the limitation on the paper space, we do not explore this framework rigorously, but intuitively, $\tau(A)$ is the maximal set of attributes shared by all the objects in A and $\varepsilon(B)$ is the maximal set of objects possessing all the attributes in B. Without loss of generality, the features or attributes can be thought of as boolean variables implying that if an object possesses a feature then the feature is *.True.* for that object otherwise it is *.False.* for that object.

In the following we present some main results on concept lattices. Let $\mathcal{C}(G, M, I)$ denote the set of all concepts of the context (G, M, I). An order relation on $\mathcal{C}(G, M, I)$ can be defined as follows. Let (A_1, B_1) and (A_2, B_2) be two concepts in $\mathcal{C}(G, M, I)$, then

$$(A_1, B_1) \leq (A_2, B_2) \quad \text{iff} \quad A_1 \subseteq A_2$$

[3] The more generalized version of this problem is of what we call *data dependency query* , which is beyond the scope of this article[3].

or equivalently $B_1 \supseteq B_2$. The concept (A_1, B_1) is called a *subconcept* of the concept (A_2, B_2) and (A_2, B_2) a *superconcept* of (A_1, B_1). Let $\mathcal{L}(G, M, I) = (\mathcal{C}(G, M, I), \leq\ )$. The fundamental theorem of Wille on concept lattices states that $\mathcal{L}(G, M, I)$ is a complete lattice called the *concept lattice* of the context (G, M, I).

If we are interested in generating all associations between B_i's (i.e., $B_i\, R\, B_j$ where $i \neq j$, and $1 \leq i, j \leq n$), then concept lattice in the context of r_2 satisfies our interest. The lattice structure corresponding to this context is given in Figure 1 in which each link gives either a trivial ($\alpha\ R\ \beta$, where $\beta \subseteq \alpha$) or a non-trivial association like ($\alpha\ R\ \beta$, where $\beta \not\subseteq \alpha$). To investigate the use of concepts in a given context for associations we give the following definitions.

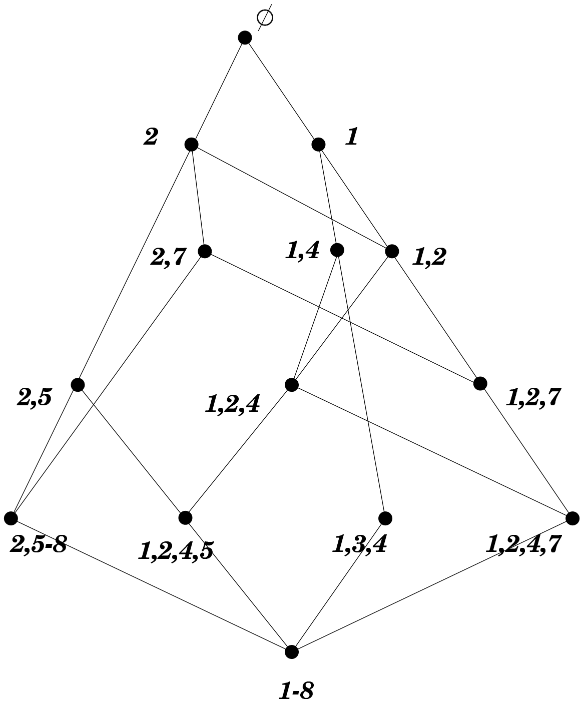

Figure 1: Concept Lattice for Basket Context

Let a concept lattice be represented by $C =< V, R >$, where V and R are set of vertices and edges, respectively. Let $a : V \rightarrow \Im(P)$ be a function yielding a subset of attributes associated with $v \in V$. We define an index set J on the vertices of V. Furthermore, we attach frequency information to each vertex by means of a function $fr : V \rightarrow N^+$ such that it returns frequency of attributes in Basket relation for a given vertex.

Each link in $C(R)$ gives us an association from $a(v_i)$ to $a(v_j)$, where $i, j \in J$. It is easy to see that the association relationship R defines a partial order on C, i.e., reflexive, transitive, and antisymmetric.

Now we are ready to explore the characteristics of R with respect to confidence and support measures.

1. Suppose $\alpha = a(v_i)$ and $\beta = a(v_j)$ for $i, j \in J$ and $\alpha R \beta \in C(R)$. Then $c(\alpha R \beta) = fr(\alpha \cup \beta)/fr(\alpha)$ with the assumption that $fr(\alpha) \neq 0$. Notice that $c(\alpha R \beta) = 1$ for $\beta \subseteq \alpha$.

2. Suppose $\alpha = a(v_i)$ and $\beta = a(v_j)$ for $i, j \in J$, and there exists no direct link connecting v_i and v_j. The confidence of $\alpha R \beta$ is computed through supremum vertex of $a^{-1}(\alpha)$ and $a^{-1}(\beta)$, say v_m. That is, $c(\alpha R \beta) = c(\alpha R \gamma) * c(\gamma R \beta)$, where $\gamma = a(v_m)$ and "*" is supremum operator.

3. Suppose $\alpha \subseteq a(v_i)$ and $\beta \subseteq a(v_j)$ for some $i, j \in J$. We postulate that $c(\alpha R \beta) = 1$ if there exists no vertex v in $C(V)$ such that $\alpha \subseteq a(v)$ and $\beta \not\subseteq a(v)$. That is, if α is always subsumed by β in C then the confidence on $\alpha R \beta$ is certain.

The rules given above are exhaustive enough to compute the confidence of any association rule that can be exposed using the Basket context. Let $\alpha \subseteq a(v_i)$ and $\beta \subset a(v_j)$. To measure the support for the association relationship $\alpha R \beta$ we propose the following rules.

1. Assume there exists a direct link connecting v_i and v_j, that is, this association is readily available from the C. Then $s(\alpha R \beta) = Pr(\alpha \cup \beta) = fr(\alpha \cup \beta)/fr(\varnothing)$.

2. In case the association is not readily available, the supremum vertex v_m of v_i and v_j, where $i, j, m \in J$, provides a basis for computing the support measure, that is, $s(\alpha R \beta) = fr(\gamma = a(v_m))/fr(\varnothing)$.

Example: Let us consider the concept lattice of the Basket relation in Figure 1. To compute the confidence on the relationship $2R\{1, 2\}$ we use the first rule, that is, $c(2R\{1, 2\}) = Pr(\{1, 2\}|2) = fr(\{1, 2\})/fr(2) = 4/5$. The support measure $s(2R\{1, 2\})$ is equal to $fr(\{1, 2\})/fr(\varnothing) = 4/6$. On contrary to the previous example, in case we look for the confidence measure related to an association relationship which do not have a direct link in C we use the second rule. Consider $1R2$. We see that the vertex containing the attributes $\{1, 2\}$ is a supremum vertex for those two vertices which contain the attributes 1 and 2, respectively. Then $c(1R2) = c(1R\{1, 2\}) * c(\{1, 2\}R2) = 4/5 * 1 = 0.8$. The support measure of $1R2$ is the same as that of $2R\{1, 2\}$ since both have the same supremum vertex. When we consider the association from the attribute 3 to the attribute 1, we find out that the product 3 is always subsumed in the product 1, that is the product 3 always goes with the one 1. This certain trend is implied by the third rule given above due to the fact that there is no vertex in C such that it contains 3, but not 1. On the other hand, the support measure of the association $3R1$ is equal to $fr(\{1,3,4\})/fr(\varnothing) = 1/6$.

We introduce the notion of a viewpoint in order that the concept lattice can be pruned to just contain useful or relevant associations [5]. A viewpoint relative to a particular subset of P will show associations that contain the given subset and those included in the subset. The viewpoint is also quite useful for association mining algorithms that perform filtering step as a priori technique to eliminate low-support associations. The viewpoint for the set {1,2,4} is shown in Figure 2.

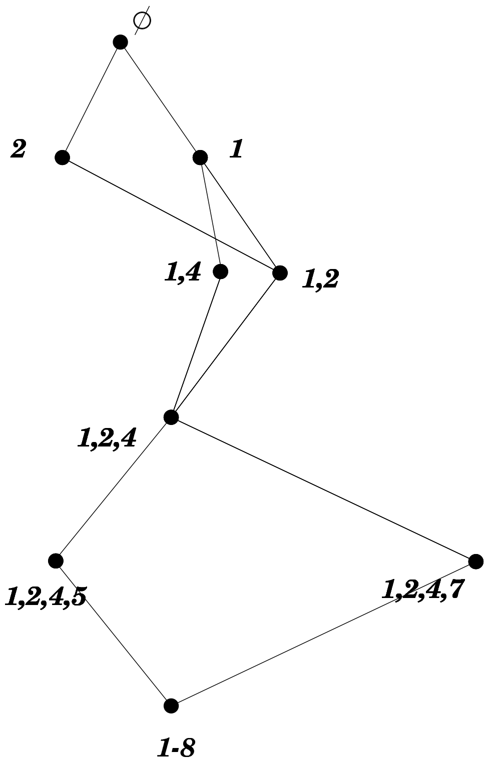

Figure 2: View of Dependencies for Products 1,2,4

3 Conclusion

Several problems remain to be investigated. One of the interesting questions is of when do we use the infimum of certain vertices? For example, we can use the infimum of (B_i and B_j) if we are interested in $Pr(B_i \cap B_j)$. Is it interesting? It might be when we look for the cases where $(B_i \cap Bj)\ R\ B_k$.

We have extended the concept analysis framework to the association problem; we believe such framework provides a natural basis for complexity analysis of the association problem, and it is general enough to evaluate several approaches to association mining. In addition to providing a viewpoint of the set for filtering some items with low-support, the concept analysis framework allows us to incrementally update the concept lattice as new transactions arrive, which cause the context Basket relation to be updated.

References

[1] R. Meo, G. Psaila, and S. Ceri, “A new SQL-like operator for mining association rules,” in *Proceedings of 22nd VLDB Conference*, (Mumbai(Bombay), India), pp. 1–12, 1996.

[2] R. Srikant and R. Agrawal, “Mining quantative association rules in large relational tables,” in *SIGMOD’96*, (Montreal, Canada), pp. 1–12, 1996.

[3] J. S. Deogun, V. V. Raghavan, A. Sarkar, and H. Sever, “Data mining: Research trends, challenges, and applications,” in *Roughs Sets and Data Mining: Analysis of Imprecise Data* (T. Y. Lin and N. Cercone, eds.), (Boston, MA), pp. 9–45, Kluwer Academic Publishers, 1997.

[4] R. Wille, “Restructuring lattice theory: An approach based on hierarchies on concepts,” in *Ordered Sets* (I. Rival, ed.), Dordrecht-Boston: Reidel, 1982.

[5] J. S. Deogun, V. V. Raghavan, and H. Sever, “Formal concept analysis and applications,” Tech. Rep. TR-CSE-98-15, University Of Nebraska at Lincoln, The Department of Computer Science, 1998.