Knowledge and Information Systems (2001) 3: 319–337
© 2001 Springer-Verlag London Ltd.

Knowledge and
Information Systems

Generalized Affinity-Based Association Rule Mining for Multimedia Database Queries

Mei-Ling Shyu[1], Shu-Ching Chen[2] and R. L. Kashyap[3]

[1]Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL, USA
[2]School of Computer Science, Florida International University, Miami, FL, USA
[3]School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA

Abstract. The recent progress in high-speed communication networks and large-capacity storage devices has led to a tremendous increase in the number of databases and the volume of data in them. This has created a need to discover structural equivalence relationships from the databases since queries tend to access information from structurally equivalent media objects residing in different databases. The more databases there are, the more query-processing performance improvement can be achieved when the structural equivalence relationships are automatically discovered. In response to such a demand, association rule mining has emerged and proven to be a highly successful technique for discovering knowledge from large databases. In this paper, we explore a generalized affinity-based association rule mining approach to discover the quasi-equivalence relationships from a network of databases. The algorithm is implemented and two empirical studies on real databases are conducted. The results show that the proposed generalized affinity-based association rule mining approach not only correctly exploits the set of quasi-equivalent media objects from the databases, but also outperforms the basic association rule mining approach in the discovery of the quasi-equivalent media object pairs.

Keywords: Association rule mining; Databases; Data mining; Knowledge discovery in databases (KDD), Multimedia

1. Introduction

In the last decade, the exponential growth of computer networks and data-collection technology, such as bar-code scanners in business domains and sensors in scientific and industrial domains, has generated an incredibly large offering of products and services for the users of computer networks. In business, data

Received 16 Sep 1999
Revised 12 Sep 2000
Accepted 9 Jan 2001

capture information such as sales opportunities and quality/cost control to improve corporate profitability. In science, data represent study observations and phenomena. In manufacturing, data help to identify performance and optimization opportunities and to improve troubleshooting processes. With the explosive growth in the amount and complexity of data, advanced data storage technology and database management systems have increased our capabilities to collect and store data of all kinds. Enterprises increasingly store and organize the huge amounts of data in data warehouses for decision-support purposes. However, our ability to interpret and analyze the data is still limited, creating an urgent need to accelerate discovery of information in databases. This need has been recognized by researchers in different areas such as database management systems (Elmasri and Navathe, 1994; Date, 1995), data warehousing (Inmon, 1992; Poe, 1996), machine learning and artificial intelligence (Shavlik and Dietterich, 1990; Langley, 1996), statistics (Elder and Pregibon, 1996), and data visualization (Lee et al., 1995; Simoudis et al., 1996). Therefore, *knowledge discovery in databases* (*KDD*) and/or *data mining* have emerged to extract useful information from the databases.

KDD is a non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data, and *data mining* is the application of algorithms for extracting patterns from data (Fayyad et al., 1996). In other words, data mining is a component in the KDD process concerned with the means by which patterns are extracted and enumerated from the data. Traditional data analysis methods often depend on humans to deal with the data directly. However, as the volume of data increases, it is not realistic to expect human experts to analyze all the data since manual data analysis simply cannot scale to handle it. In addition, knowledge acquisition from experts may be biased and need to be validated with broader tests. *KDD* or *data mining* can help to overcome the limitations.

Data mining is a process for extracting non-trivial, implicit, previously unknown and potentially useful information from data in databases. Three of the most common methods in data mining are association rules (Srikant and Agrawal, 1995; Srikant and Agrawal, 1996), data classification (Lu et al., 1995; Cheeseman and Stutz, 1996), and data clustering (Ester et al., 1995; Zhang et al., 1996). Association rules discover the co-occurrence associations among data. Data classification is the process that classifies a set of data into different classes according to some common properties and classification models. Finally, data clustering groups physical or abstract objects into disjoint sets that are similar in some respect. By knowledge discovery in databases, interesting knowledge, regularities, or high-level information can be extracted from the relevant sets of data in databases and be investigated from different angles; large databases thereby serve as rich and reliable sources for knowledge generation and verification.

In our previous study, we proposed a probabilistic network-based mechanism to facilitate the functionality of a *multimedia database management system* (MDBMS) (Shyu et al., 1998a, 1998b). With the help of probabilistic networks, methods can be developed to discover useful information and knowledge for the multimedia databases via probabilistic reasoning. Multimedia databases are considered since each multimedia database includes not only images, audio, graphics, animation, and full-motion video, but also text as in traditional text-based databases. In addition, data access and manipulation for multimedia databases are more complicated than those of the conventional databases since it needs to incorporate diverse media with diverse characteristics. Since the primitive constructed or manipulated entities in most multimedia systems are called *media objects* (Candan et al., 1998), a *media object* is used as a basic unit in our mechanism. Moreover, since each media object is associated with an *augmented transition network* (ATN) which models multimedia presentations, multimedia database searching, and multimedia browsing (Chen and Kashyap, 1997; Chen and Kashyap, 1999), our mechanism has the capabilities to query different media types and manage the rich semantic multimedia data for multimedia databases.

In this paper, we explore a new data-mining capability that involves mining quasi-equivalence relationships from a network of databases to enhance our probabilistic network-based mechanism (Shyu et al., 1999). Because of the navigational characteristics, queries tend to access information from related or structurally equivalent media objects which span multiple multimedia databases. Since a database schema represents a non-redundant view, media object equivalence cannot exist in a single database. Therefore, only media objects across different databases can be structurally equivalent. Two media objects are said to be equivalent if they are deemed to possess the same real world states (RWS's) (Navathe et al., 1986; Larson et al., 1989), i.e., if these two media objects represent the same sets of instances of the same real-world entity. For example, a database contains a media object $EMPLOYEE$ with attributes *name*, *id*, *address*, *department*, and *salary*. Another database has a media object EMP, representing the enrollment of employees in training courses and containing attributes *name* and *courses*. $EMPLOYEE$ and EMP in these two databases should represent the same RWS's for the organization so that they are structurally equivalent. Here, the quasi-equivalent relationship is used to approximate the structurally equivalent relationship.

As the number of databases and the volume of the data increase, query processing performance depends heavily on the capability to discover the structural equivalence relationships of the media objects from the network of databases. For this purpose, a generalized affinity-based association rule mining approach that discovers the set of quasi-equivalent media objects from databases is proposed. Association rule mining has recently attracted strong attention and proven to be a highly successful technique for extracting useful information from very large databases. Intuitively, associated items appear together frequently. Discovering associations in a database will uncover the affinities among the collection of data in the database. These affinities between data are represented by association rules. We use the relative affinity measures to indicate how frequently two media objects are accessed together. The calculations of support, confidence, and interest for association rules are based on the relative affinity values. The proposed affinity-based approach provides more informative feedback since the relative affinity measures consider the access frequencies of queries and can incorporate into current item set algorithms with no decrease in efficiency.

The generalized affinity-based association rule mining process consists of two phases. Phase I iteratively checks a set of constraints: (1) minimum interest threshold, (2) interest constraint, and (3) refinement constraint. In Phase II, a minimum confidence threshold constraint is first checked and then some further conditions can be imposed if any unreasonable situation exists. The algorithm is implemented and two empirical studies on real database management systems at Purdue University are conducted. The first study is to empirically test the proposed generalized affinity-based association rule mining approach. The second study is to compare the performance of the proposed association rule mining algorithm with the basic association rule mining approach. The results from the empirical studies show that the proposed algorithm discovers the set of quasi-equivalence media objects for the databases to assist in enhancing query processing performance, and outperforms the basic association rule mining approach in discovering the quasi-equivalence relationships.

This paper is organized as follows. In Section 2, the discovery of association rules and the formalization of the affinity-based association rules are presented. The proposed *generalized affinity-based association rule mining* algorithm is given in Section 3. Two empirical studies to test the proposed algorithm and to compare the proposed algorithm with the basic association rule mining approach are conducted in Section 4. Section 5 concludes this paper.

2. Discovery of Association Rules

In this section, the *support*, *confidence*, and *interest* measures for the basic association rule mining approach and the proposed generalized affinity-based association rule mining approach are introduced.

2.1. Basic Association Rules

One of the most important problems in data mining is the discovery of association rules for large databases. Association rules are a simple and natural class of database regularities. The purpose is to discover the co-occurrence associations among data in large databases, i.e., to find items that imply the presence of other items in the same transaction. Association discovery was first introduced by Agrawal et al. (1993). Given a set of transactions, where each transaction contains a set of items, an association rule is defined as an expression $X \rightarrow Y$, where X and Y are sets of items and $X \cap Y = \emptyset$. The rule implies that the transactions of the database which contain X tend to contain Y.

There are three measures of the association: *support*, *confidence* and *interest*. The support factor indicates the relative occurrence of both X and Y within the overall data set of transactions and is defined as the ratio of the number of tuples satisfying both X and Y over the total number of tuples. The confidence factor is the probability of Y given X and is defined as the ratio of the number of tuples satisfying both X and Y over the number of tuples satisfying X. In other words, the support factor indicates the frequencies of the occurring patterns in the rule, and the confidence factor denotes the strength of implication of the rule (Chen et al., 1996). Since not all the discovered association rules that pass the minimum support and minimum confidence factors are interesting enough to present, sometimes an interest factor is defined to indicate the usefulness of the rules. The interest factor is a measure of human interest in the rule. For example, a high interest means that if a transaction contains X, then it is much more likely to have Y than the other items.

Let N to be the total number of tuples and $\mid A \mid$ to be the number of tuples containing all items in the set A. Define

$$support(X) = P(X) = \frac{\mid X \mid}{N} \tag{1}$$

$$support(X \rightarrow Y) = P(X \cup Y) = \frac{\mid X \cup Y \mid}{N} \tag{2}$$

$$confidence(X \rightarrow Y) = \frac{P(X \cup Y)}{P(X)} = \frac{|X \cup Y|}{|X|} \tag{3}$$

$$interest(X \rightarrow Y) = \frac{P(X \cup Y)}{P(X)P(Y)}. \tag{4}$$

The problem is to find all the association rules satisfying user-specified minimum support and minimum confidence constraints that hold in a given database. Rules with high support and confidence factors represent a higher degree of relevance than rules with low support and confidence factors.

2.2. Affinity-Based Association Rules

In this paper, a relative affinity value between two media objects is used to measure how frequently these two media objects have been accessed together in a set of queries (Shyu et al., 1998a). Here, the set of queries is considered as the set of transactions since, similar to the case that each transaction may contain one or more items, each issued query may request information from one or more media objects from the databases. In addition, an item may be purchased in multiples in a transaction, which can be thought of as having a weight in the transaction. However, the current definition of *support* tells only the number of transactions containing an item set but not the number of items. In this case, an item with a larger weight should be considered more frequently than the original *support* measure indicates. In order to allow the *support* measure to be able to capture the actual frequencies of the occurring patterns in the rule, the items should be given weights in the calculation of the *support* measure. Similarly, each query could have a distinct frequency, i.e., a query may be activated several times. Again, the query access frequency can be thought as the weight of the query. For example, though the number of outcomes that two media objects are accessed by the same queries is small, if the total access frequency of those queries accessing both of them is high, then the relative affinity between these two media objects is considered to be high. Therefore, the actual access frequency of a query per time period should be taken into account when the relative affinity between two media objects is calculated, and the calculations of *support*, *confidence*, and *interest* for association rules are based on the relative affinity values. Using the relative affinity measures allows more informative feedback because it tells the number of accesses of the queries but not the number of queries.

A set of historical data which includes the query access frequencies and the usage patterns is provided as the prior information for the proposed approach. In a database management system, the access patterns of the media objects of the queries and the access frequencies of the queries can be collected and recorded in a log file. Let Q = {1, 2, ..., q} be the set of sample queries that run on the multimedia databases $d_1, d_2, \ldots, d_p$ with media object set OC = {1, 2, ..., g} in the multimedia database system. Also, let *m* and *n* be two media objects. Define the variables:

- $use_{k,m}$ = usage pattern of media object *m* with respect to query *k* per time period (available from the historical data)

$$use_{k,m} = \begin{cases} 1 & \text{if media object } m \text{ is accessed by query } k \\ 0 & \text{otherwise} \end{cases}$$

- $access_k$ = access frequency of query k per time period (available from the historical data)
- $aff_{m,n}$ = relative affinity measure of media objects m and n

The $use_{k,m}$ has value 1 if media object m is accessed by query k and value 0 otherwise. The values for $access_k$ and $use_{k,m}$ are available from the set of historical data. An example set of historical data can be found in Shyu et al. (1998a). Based on the above variable definitions, we define the affinity-based support, confidence, and interest factors for the association rules as follows:

$$aff_{m,n} = \sum_{k=1}^{q} use_{k,m} \times use_{k,n} \times access_k \tag{5}$$

$$support(m) = \frac{\sum_{k=1}^{q} use_{k,m} \times access_k}{\sum_{k=1}^{q} access_k} \tag{6}$$

$$support(m \rightarrow n) = \frac{aff_{m,n}}{\sum_{k=1}^{q} access_k} \tag{7}$$

$$confidence(m \rightarrow n) = \frac{support(m \rightarrow n)}{support(m)} \tag{8}$$

$$interest(m \rightarrow n) = \frac{support(m \rightarrow n)}{support(m)support(n)}. \tag{9}$$

Here, *support*(m) indicates the fraction of the number of accesses of the media object m with respect to the total number of accesses for all the queries. The *support* value of the rule ($m \rightarrow n$) shows the probability of accessing both media objects m and n with respect to all the accesses of the queries. The *confidence* value of the rule ($m \rightarrow n$) denotes the probability of accessing media object n given that media object m has been accessed for the queries. The *interest* value of the rule ($m \rightarrow n$) gives the measurement that if media object m is accessed by a query, then media object n is much more (or much less) likely to be accessed by the same query. For example, a high interest value of the rule ($m \rightarrow n$) implies that media object n is much more likely to have a high-affinity relationship with m than other media objects. Then, these values are used in the proposed generalized affinity-based association rule mining algorithm to find the set of quasi-equivalent media objects. The quasi-equivalent relationship is used to approximate the structurally equivalent relationship. Moreover, since we try to discover the quasi-equivalence relationship of two media objects, only the 2-item sets are considered at the current stage. Hence, the overheads such as database scans and large item set generations can be reduced. We plan to extend the framework to discover the quasi-equivalent relationships for larger item sets (if any) in the future.

3. The Generalized Affinity-Based Association Rule Mining

In this section, the generalized affinity-based association rule mining that discovers a set of quasi-equivalent media objects in a network of databases is proposed. Since queries tend to access information from related or structurally equivalent media objects residing across multiple databases in an information-providing environment, the discovery of the structural equivalence relationships

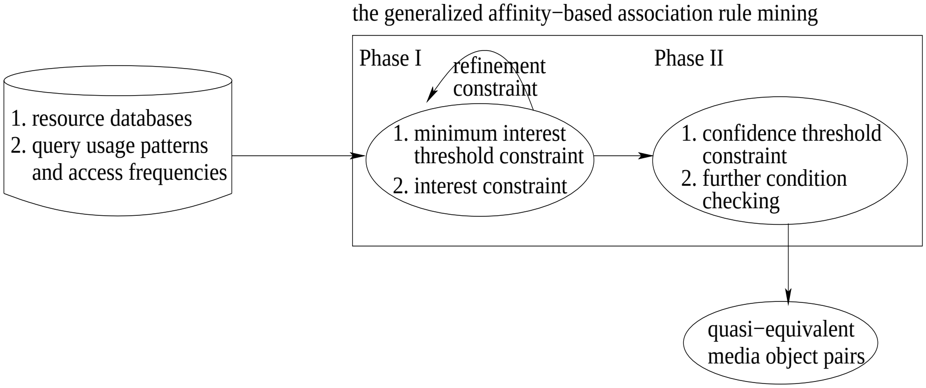

the generalized affinity−based association rule mining
Phase I
refinement constraint
Phase II
1. resource databases
2. query usage patterns and access frequencies
1. minimum interest threshold constraint
2. interest constraint
1. confidence threshold constraint
2. further condition checking
quasi−equivalent media object pairs

Fig. 1. Architecture for the generalized affinity-based association rule mining algorithm.

is very critical in improving query processing performance. For example, a given database might contain a media object $EMPLOYEE$, given attributes *name*, *id*, *address*, *department*, and *salary*. Another database has a media object EMP file, representing the enrollment of employees in training courses and containing attributes *name* and *courses*. These two media objects are structurally equivalent since they represent the same RWS's for the organization. Suppose that in order to carry out the process of training course administration, it is necessary to know the department for each enrolled employee. To answer this type of query, it is required to access information from both media objects. Hence, if the knowledge such as the structural equivalence relationship between these two media objects can be discovered in advance automatically, query processing performance can be greatly enhanced.

3.1. Architecture

Figure 1 shows the architecture for the *generalized affinity-based association rule mining algorithm.* As can be seen from Fig. 1, the multimedia resource databases, the query usage patterns, and the query access frequencies are the inputs for the proposed association rule mining algorithm. The main task of the association rule mining algorithm is to discover the set of quasi-equivalent media objects which can be used to assist in improving query processing performance. There are two major phases for the *generalized affinity-based association rule mining* process. Phase I is executed iteratively based on the refinement of the minimum interest threshold to generate the candidate set of quasi-equivalent media objects until a predefined refinement constraint is met. Then, based on the candidate set generated from Phase I, Phase II checks the minimum confidence threshold and further conditions (if any) to get the final set of quasi-equivalent media objects.

There are several parameters required in both phases. The values for $cria1$, $cria2$, and $Conf$ need to be decided by the users before the algorithm is run.

- I_m is the maximal interest value for each media object m. I_m is obtained by finding the maximal $interest(m \rightarrow n)$ value where a media object n is in a different database since the equivalence relationship can occur only when two media objects are from different databases.

- *IntTd* is the minimum interest threshold. It is defined as 'iteration number × *cria*1 × I_m', where *cria*1 is a criterion value. Hence, the minimum interest threshold increases as the number of iteration increases.
- The refinement constraint threshold is defined to be '*cria*2 × the total number of media objects', where *cria*2 is a criterion value.
- *Conf* is the minimum confidence threshold.

We now roughly discuss the steps for the two phases. The detailed algorithm will be introduced in the next subsection. Phase I starts with a set of constraints: (1) minimum interest threshold, (2) interest constraint, and (3) refinement constraint. Any pair whose association rule has an interest value exceeding the interest threshold is first selected into the candidate pool. Next, the interest constraint is imposed to shrink the size of the candidate pool: the pair (m, n) remains in the candidate pool only if both (m, n) and (n, m) are in the candidate pool. That is, both $interest(m \rightarrow n)$ and $interest(n \rightarrow m)$ must satisfy the interest threshold criterion to make sure they are interesting enough in both directions. Then, the output of Phase I consists of a list of pairs of candidates. On seeing the candidates, the refinement constraint is checked to see whether further interest threshold refinement is necessary or not. In this manner, Phase I is iterative. Once satisfied with the current candidate list, the process proceeds to Phase II, wherein two constraints are set: (1) minimum confidence threshold, and (2) whatever further conditions to be imposed. The minimum confidence threshold is used again to cut down the candidate pool size. The pair (m, n) stays in the candidate pool if either $confidence(m \rightarrow n)$ or $confidence(n \rightarrow m)$ reaches the minimum confidence threshold. Upon examining the output, further conditions can be imposed to get rid of unreasonable pairs in the candidate pool.

The reason for having the refinement constraint is to avoid setting the minimum interest threshold value too high. If the value is set too high, then lots of possible candidate media object pairs may not be included in the candidate pool at the first and/or the second constraint checkings in Phase I. In addition, the refinement constraint is set since the algorithm currently considers only the rules with two media objects, and the fact that two databases usually have only one equivalence relationship if there is any. Hence, the refinement constraint is used to refine the candidate pool by increasing the minimum interest threshold value. The refinement constraint makes Phase I iterative. In order to clarify the iterative steps with the non-iterative steps, the algorithm is separated into two phases.

3.2. Algorithm

In this subsection, the proposed *generalized affinity-based association rule mining* algorithm that discovers the set of quasi-equivalent media object pairs is introduced. This mining process is very useful for exploring some semantic relationships from the complicated data structures of the databases automatically, and requires parameters such as the minimum interest threshold, refinement constraint, and minimum confidence threshold to be determined by the users subjectively according to different requirements for different applications. This flexibility allows users to set the criteria suitable for different applications. Though the mining process is used to find the set of quasi-equivalent media objects in this paper, it can also be used in other applications. For example, in manufacturing, there exist hundreds of assembly–subassembly part relationships (Rosenthal and Heiler, 1987). These relationships correspond to the concept of 'composition' defined in the OSAM data model in Su et al. (1989) or the *aggregation* relationships. An aggregation hierarchy expresses part-of relationships between two media objects with 1:M cardinality by definition. Media objects are organized into an aggregation structural hierarchy if one media object is composed by other media objects in a nested or hierarchical fashion. This mining process can be applied to exploit some of the semantic relationships such as the assembly–subassembly part semantic relationships for the applications in the manufacture domain. Of course, the definitions of the affinity, support, confidence, and interest, and the selections of the parameters need to be adjusted accordingly.

Here, the details of the algorithm for the *generalized affinity-based association mining* process are introduced. Start with all the media objects in the databases. Let L_1 and L_2 represent the sets of 1-item sets and 2-item sets, where each 1-item set has one media object and each 2-item set has two media objects. Generate L_2 by $L_1 * L_1$ where $*$ is an operation for concatenation. The algorithm needs to make only one pass over the database. While the only pass is made, one record at a time is read and $support(m)$, $aff_{m,n}$, and the summation of $access_k$ are computed. After that, $support(m \rightarrow n)$, $interest(m \rightarrow n)$, and $confidence(m \rightarrow n)$ can be obtained. There is no need to do multiple database scans, thus reducing the processing overheads.

We now discuss how to generate the candidate pool and how to determine the set of quasi-equivalent media objects. Let the number of media objects in the databases be Nmo and the resulting set be candidate_pool.

Steps for Phase I.

```
1. For all the 1-item sets, compute $support(m)$ (equation (6)).
2. For all the 2-item sets,
   • Compute $aff_{m,n}$ (equation (5)).
   • Compute $support(m \rightarrow n)$ (equation (7)).
   • Compute $confidence(m \rightarrow n)$ (equation (8)).
   • Compute $interest(m \rightarrow n)$ (equation (9)).
3. Initialize candidate_pool = $\emptyset$ and $iter = 1$; set the values for $cria1$ and $cria2$.
4. For $m = 1$ to $Nmo$,
   (a) If $iter = 1$, then find the maximal interest value $I_m$.
   (b) Set the minimum interest threshold $IntTd = cria1 \times iter \times I_m$.
   (c) For those media objects $n$'s,
       if $iter = 1$ and $interest(m \rightarrow n) \geqslant IntTd$,
       then candidate_pool = candidate_pool $\bigcup \{(m,n)\}$.
       else if $interest(m \rightarrow n) < IntTd$,
       then $(m,n)$ is removed from candidate_pool.
5. Check the interest constraint:
   if $(m,n) \in$ candidate_pool and $(n,m) \notin$ candidate_pool,
   then $(m,n)$ is removed from candidate_pool.
```

6. Check the refinement constraint:
   if the number of media objects which have zero or one pair remaining in the candidate_pool $\geqslant cria2 \times Nmo$,
   then goto Phase II.
   else set $iter = iter + 1$ and goto step 4.

The first step is to compute the *support*(m) for every 1-item set using equation (6). Since each query may be activated multiple times, the actual access frequency of each query is taken into account in calculating *support*(m) and *support*($m \rightarrow n$) values. That is why this mining process is *affinity-based*. The advantage of using the relative affinity measures is to allow more informative feedback because it tells the number of accesses of the queries but not the number of queries. The second step is to compute the $aff_{m,n}$, *support*($m \rightarrow n$), *confidence*($m \rightarrow n$), and *interest*($m \rightarrow n$) using equations (5, 7, 8, 9) for all the media object pairs. Only the *interest*($m \rightarrow n$) and *confidence*($m \rightarrow n$) values are needed in determining the set of quasi-equivalent media objects. In the third step, the candidate_pool is initialized as an empty set and the number of iteration (*iter*) is set to one. Also, the values for the minimum interest threshold (*cria*1) and the refinement constraint (*cria*2) need to be defined. Again, these criteria can be adjusted for different applications. Step 4 executes a for-loop for all the media objects. First, the maximal interest values for all the media objects on the first iteration are found. Once the maximal interest value I_m for media object m is obtained, the minimum interest threshold can be calculated according to the predefined formula. Similarly, the formula to calculate the minimum interest threshold can be varied for different applications. Then, the corresponding media object pair is put into the candidate_pool or removed from the candidate_pool by comparing its interest value with the minimum interest threshold. The candidate_pool constructed from step 4 goes to step 5 for the interest constraint checking. Since only those media object pairs whose interest values are above the minimum interest threshold on both directions are interesting enough to be considered as quasi-equivalent, the interest constraint is used to cross out the unsatisfied pairs from the candidate_pool in step 5. In step 6, the refinement constraint is checked to see whether another iteration is required. If the number of the media objects which have zero or one pair remaining in the candidate_pool is equal to or greater than the refinement constraint, then Phase I stops and goes to Phase II. Otherwise, it goes to step 4 for another iteration.

Steps for Phase II.

1. Set the minimum confidence threshold $Conf$.
2. For each pair (m, n) in candidate_pool,
   if $confidence(m \rightarrow n) < Conf$ and $confidence(n \rightarrow m) < Conf$,
   then (m, n) is removed from candidate_pool.
3. Check if further conditions need to be imposed to remove some unreasonable situations.

The steps for Phase II are used to eliminate those media object pairs that are potentially non-equivalent. First, the minimum confidence threshold needs to be defined. Again, this threshold can be adjusted accordingly for different applications. The second step is to remove those media object pairs whose confidence values are smaller than the minimum confidence threshold on both directions. However, since some situations cannot be reflected directly by the numbers of accesses from the historical data, human reasoning is required. The

Table 1. The maximal interest measure I_m for each media object m.

m	1	2	3	4	5	6	7	8
I_m	1.387	5.863	468.603	2.198	2.479	4.409	4.409	8.835
m	9	10	11	12	13	14	15	16
I_m	468.603	23.238	27.879	3.805	8.835	27.879	8.835	8.026
m	17	18	19	20	21	22		
I_m	1.837	23.238	2.861	3.805	4.409	2.479		

last step of Phase II is to check whether there exist some unreasonable situations in the candidate_pool. For example, a media object cannot have equivalent relationships with two or more media objects in the same database at the same time since equivalence can only occur for media objects in different databases. These unreasonable situations need to be examined by humans to get the final set of quasi-equivalent media object pairs.

4. Empirical Studies

Two empirical studies on the financial database management systems at Purdue University in July, August, and September for the year 1997 were conducted. The databases represent 22 media objects accessed by 17,222 queries. Let the media objects be numbered from 1 to 22 and the media objects in the same database have consecutive numbers. The first study empirically tests the proposed generalized affinity-based association rule mining approach. The second study compares the performance of the proposed association rule mining algorithm with the basic association rule mining approach.

The basic association rule mining approach and the proposed generalized affinity-based association rule mining approach are implemented using the C++ programming language. The differences between the implementation of these two approaches are mainly in the equations for $support(m)$ and $support(m \rightarrow n)$. In the basic approach, $support(m)$ is the value of the number of queries accessing media object m divided by the total number of queries, and $support(m \rightarrow n)$ is the ratio of the number of queries that the media objects m and n are both accessed over the total number of queries. On the other hand, in the affinity-based approach, $support(m)$ indicates the fraction of the number of accesses of media object m with respect to the total number of accesses for all the queries, and the *support* value of the rule $(m \rightarrow n)$ shows the probability of accessing both media objects m and n with respect to all the accesses of the queries, where the number of query accesses take into account the access frequency of each query.

4.1. Empirical Study One

We implemented the proposed association rule mining algorithm with the affinity-based support, confidence, and interest measures reflecting the number of accesses for each media object. Set the values for the three criteria to be $cria1 = 0.2$, $cria2 = 0.5$, and $Conf = 99\%$.

Two iterations were executed in Phase I. At the first iteration, the I_m measures for all media objects m's were first found (as shown in Table 1). Note that the maximal interest value for a media object may occur in multiple places. This situation occurs when *support*$(m \rightarrow n)$ is equal to *support*(n). That is, those queries which access media object n also access media object m. For example, the pairs (1,9) and (1,20) both have interest value 1.387, which indicates that those queries which access media object 9 also access media object 1. Similarly, those queries which access media object 20 also access media object 1. However, the maximal interest for 9 occurs at the pair (9,3) and the maximal interest for *20* occurs at the pair (20,12). From the observations, if the I_m measure occurs at *interest*$(m \rightarrow n)$, the I_n measure occurs at *interest*$(n \rightarrow m)$, and I_m equals I_n, then m and n are potentially quasi-equivalent. Since those queries which access m also access n and those queries which access n also access m, this indicates that m and n are accessed by the same set of queries and thus they are very likely to have the quasi-equivalence relationship. In addition, we observe that when the I_m measure is very large, it converges to one quasi-equivalence pair for the corresponding media object m faster than other media objects. The reason is that a certain percentage (0.2, 0.4, etc.) of the I_m value is used as the criterion to maintain the candidate_pool. When the I_m value is much larger than other interest values, it is possible that other media objects will be crossed out of the candidate_pool in one or two iterations. As can be seen from Table 1, the maximal interest value for media object 3 is 468.603 which occurs at the pair (3,9) and at the same time the maximal interest value for media object 9 is 468.603 which occurs at the pair (9,3). Since the value 468.603 is extremely larger than other interest values for media objects 3 and 9, only the pairs (3,9) and (9,3) remain in the candidate_pool for media objects 3 and 9 in the first iteration (as shown in Fig. 2(a)). Figure 2 shows the candidate pairs in the candidate_pool for each iteration and each phase for this study.

When the I_m measures are determined, the $IntTd$ for the first iteration is set to $0.2 \times I_m$ and 97 pairs are generated in the candidate_pool. After the interest constraint, 30 pairs are removed and the refinement constraint checking indicates that there is a need to go to the second iteration. The refinement constraint is to check whether the number of the media objects which have zero or one pair remaining in the candidate_pool is equal to or greater than 11 (i.e., 0.5×22). The first column in Fig. 2(a) is each individual media object and the second column lists the candidate media objects corresponding to that individual media object. Those media objects that do not meet the interest constraint in the candidate media object list are crossed out from the candidate media object list. The resulting media object list is then input to the second iteration. At the second iteration, the minimum interest threshold $IntTd$ is incremented to $0.4 \times I_m$ which makes the pool shrink to 52 pairs. Next, the interest constraint is checked and 12 pairs are removed (as shown in Fig. 2(b)). Then, the refinement constraint is satisfied so that Phase I stops and the size of the pool goes from 97 pairs down to 40 pairs. That is, more than half of the pairs have been removed after Phase I is executed. Since the interest measures are based on the affinity relationships of the media objects, saying that the association $(m \rightarrow n)$ has high interest means that if the media object m is accessed by a query, then the media object n is much more likely to be accessed by the same query than other media objects. That is, media object n is much more likely to have a high-affinity relationship with m than other media objects. Similarly, if both associations $(m \rightarrow n)$ and $(n \rightarrow m)$ satisfy the minimum interest threshold and interest constraint, then the pairs (m, n) and (n, m) are most likely to be quasi-equivalent.

In Phase II, the minimum confidence threshold *Conf* is set to be 99%. The

PHASE I

(a) candidate_pool: (iteration 1)

media object	media object list
1	7, ~~8~~, ~~9~~, ~~10~~, ~~11~~, 12, ~~13~~, ~~14~~, ~~15~~, ~~16~~, 17, ~~18~~, 19, 20, 21
2	~~9~~, 10
3	9
4	~~8~~, ~~10~~, 11, ~~14~~, 15, 16, 17, ~~18~~, 22
5	7, ~~8~~, ~~10~~, ~~11~~, 12, ~~14~~, 17, ~~18~~, 22
6	7, 17
7	1, ~~3~~, 5, 6, ~~10~~, 17, ~~18~~, 21
8	13, 15, 16, 17
9	3
10	2, ~~3~~, 18
11	14
12	1, 5, 17, 19, 20
13	8, 18
14	11
15	4, 8
16	~~3~~, 4, 8
17	1, 4, 5, 6, 7, 8, ~~10~~, ~~11~~, 12, ~~13~~, 19, 20, 21
18	~~3~~, 10, 13
19	1, ~~6~~, 12, 17
20	1, 12, 17
21	1, 7, 17
22	~~3~~, 4, 5

(b) candidate_pool: (iteration 2)

media object	media object list
1	~~7~~, ~~12~~, 17, 19, ~~20~~, ~~21~~
2	~~10~~
3	9
4	~~11~~, ~~15~~, ~~16~~, 22
5	~~7~~, 17, 22
6	7, 17
7	6, 21
8	13, 15, 16
9	3
10	18
11	14
12	17, 19, 20
13	8, ~~18~~
14	11
15	8
16	8
17	1, 5, 6, ~~7~~, ~~8~~, 12, 19, 20, 21
18	10
19	1, 12, 17
20	12, 17
21	7, 17
22	4, 5

PHASE II

(c) candidate_pool: (confidence checking)

media object	media object list
1	~~17~~, 19
2	
3	9
4	~~22~~
5	~~17~~, ~~22~~
6	7, 17
7	6, 21
8	13, 15, ~~16~~
9	3
10	~~18~~
11	14
12	~~17~~, ~~19~~, 20
13	8
14	11
15	8
16	~~8~~
17	~~1~~, ~~5~~, 6, ~~12~~, 19, 20, 21
18	~~10~~
19	1, ~~12~~, 17
20	12, 17
21	7, 17
22	~~4~~, ~~5~~

(d) candidate_pool: (further checking)

media object	media object list
1	19
3	9
6	7, ~~17~~
7	6, 21
8	13, 15
9	3
11	14
12	20
13	8
14	11
15	8
17	~~6~~, ~~19~~, ~~20~~, ~~21~~
19	1, ~~17~~
20	12, ~~17~~
21	7, ~~17~~

Fig. 2. The candidate pairs in the candidate_pool.

reason for such a high confidence threshold is that rules with high confidence factors represent a higher degree of relevance than rules with low confidence factors. Since we try to approximate the structural equivalence relationship, which requires a high confidence factor, the confidence threshold is set high for this purpose. There are 24 pairs left in the candidate_pool after the confidence constraint checking (as shown in Fig. 2(c)). Finally, it is checked whether some unreasonable situations exist and need to be avoided. In the current candidate_pool, media object numbered 17 appears to have quasi-equivalence relationships with media objects numbered 6, 19, 20, and 21. This is unreasonable because of the following two observations. First, media objects numbered 19, 20, and 21 belong to the same database. As mentioned previously, equivalence relationships exist only between two media objects from different databases. Hence, it is impossible for media object numbered 17 to be quasi-equivalent to all three of them. Second, media object numbered 6 is quasi-equivalent to media object numbered 21, and at the same time is in the same database as media object numbered 1 which is quasi-equivalent to media object numbered 19. Hence, media object numbered 17 cannot have quasi-equivalence relationships to media objects numbered 6, 19, and 21. From the above two observations, eight more pairs are removed and the final number of pairs in the candidate_pool is 16 (as shown in Fig. 2(d)). Since the quasi-equivalence relationship of the pair (m, n) is the same as the quasi-equivalence relationship of the pair (n, m), if the order of the two media objects is not considered, there are eight quasi-equivalent media object pairs after the association rule mining process.

4.2. Empirical Study Two: Comparisons

In this study, the affinity-based association rule mining algorithm (Shyu et al., 1999) and the basic association rule mining approach (Agrawal et al., 1993; Chen et al., 1996) are compared by using the same database management systems in the discovery of the quasi-equivalence relationships. For the basic approach, the *support* and *confidence* of an association rule are calculated without considering the access frequencies of the queries (i.e., not affinity-based). The $support(m \rightarrow n)$ is the ratio of the number of queries that the media objects m and n are both accessed over the total number of queries. The $confidence(m \rightarrow n)$ is the ratio of the number of queries that m and n are both accessed over the number of queries that m is accessed.

Table 2 lists all the media object pairs that satisfy the corresponding support and confidence values under the basic association rule mining approach. As can be seen from this table, the number of media object pairs decreases when the support value increases. For example, there are many media object pairs satisfying the condition when the support value is from 10% to 30%. However, when the support value is set to 40% or 50% and the confidence value ranges from 10% to 99%, only less than two media object pairs remain in the table. There is even no media object pair satisfying the condition when the support value is greater than or equal to 60%.

Moreover, the number of media object pairs also decreases as the confidence value increases under the same support value. For example, when the support value is set to 10%, there are 18 media object pairs that have confidence value greater than or equal to 10%, 14 media object pairs that have confidence value from 20% to 40%, 11 media object pairs that have confidence value 50%, etc.

Table 2. The media object pairs satisfying various support and confidence values for the basic association rule mining approach. The numbers in the first column are the various **support** values; while the numbers in the first row are the various **confidence** values.

	Confidence									
	10%	20%	30%	40%	50%	60%	70%	80%	90%	99%
10%	(1,4)	–	–	–	–	–	–	–	–	–
	(1,7)	–	–	–	–	–	–	–	–	–
	(1,12)	(1,12)	(1,12)	(1,12)	–	–	–	–	–	–
	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	–	–	–
	(1,20)	(1,20)	(1,20)	(1,20)	–	–	–	–	–	–
	(4,1)	–	–	–	–	–	–	–	–	–
	(7,1)	(7,1)	(7,1)	(7,1)	(7,1)	(7,1)	(7,1)	(7,1)	–	–
	(7,17)	(7,17)	(7,17)	(7,17)	(7,17)	(7,17)	–	–	–	–
	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	–
	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	–
	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	–	–
	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	–
	(17,7)	–	–	–	–	–	–	–	–	–
	(17,12)	(17,12)	(17,12)	(17,12)	(17,12)	–	–	–	–	–
	(17,20)	(17,20)	(17,20)	(17,20)	–	–	–	–	–	–
	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)
	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)
	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)
20%	(1,12)	(1,12)	(1,12)	(1,12)	–	–	–	–	–	–
	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	–	–	–
	(1,20)	(1,20)	(1,20)	(1,20)	–	–	–	–	–	–
	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	–
	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	–
	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	–	–
	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	–
	(17,12)	(17,12)	(17,12)	(17,12)	(17,12)	–	–	–	–	–
	(17,20)	(17,20)	(17,20)	(17,20)	–	–	–	–	–	–
	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)
	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)
	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)
30%	(1,12)	(1,12)	(1,12)	(1,12)	–	–	–	–	–	–
	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	–	–	–
	(1,20)	(1,20)	(1,20)	(1,20)	–	–	–	–	–	–
	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	(12,1)	–
	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	(12,17)	–
	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	(12,20)	–	–
	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	–
	(17,12)	(17,12)	(17,12)	(17,12)	(17,12)	–	–	–	–	–
	(17,20)	(17,20)	(17,20)	(17,20)	–	–	–	–	–	–
	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)	(20,1)
	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)	(20,12)
	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)	(20,17)
40%	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	–	–	–
	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	–
50%	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	(1,17)	–	–	–
	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	(17,1)	–
60%	–	–	–	–	–	–	–	–	–	–

Though there are many media object pairs when the support values range from 10% to 30%, there are very few media object pairs that satisfy a high confidence value. In our proposed affinity-based approach, the minimum confidence threshold *Conf* is set to be 99% since rules with high confidence values represent a higher degree of relevance than rules with low confidence values, and the quasi-equivalence relationship requires a high confidence value. If the same confidence value is required, then only the media object pairs (20,1), (20,12), and (20,17) under the support value 10%, 20%, or 30% can be selected. Even if these three media object pairs satisfy the conditions, only (20,12) is actually a structurally equivalent media object pair. In addition, when the support value is above 30% and the confidence value is 99%, no media object pair satisfies both conditions. From the observations, it is easy to see that the majority of the media object pairs on Table 2 do not have the structural equivalence relationships even if they satisfy both conditions. Under the basic association rule mining approach, only one media object pair is correctly discovered as being structurally equivalent, and the rest of the media object pairs do not match with the correct structurally equivalent media object pairs. In other words, while the basic association rule mining approach discovers the incorrect quasi-equivalent media object pairs, it does not discover the correct structurally equivalent media object pairs.

One of the reasons that the proposed affinity-based association rule mining algorithm outperforms the basic association rule mining approach is the inclusion of the query access frequencies in the calculations of the *support* measures. In the basic approach, the definition of *support* reflects only the number of queries accessing two media objects but not the access frequencies of the queries. Though the number of outcomes that two media objects are accessed by the same queries is small, if the total access frequencies of those queries accessing both of them is high, then the relative affinity between these two media objects should be considered high. By incorporating the access frequencies of the queries into the calculation of the support value, more realistic affinity relations can be captured to reflect the association relations. Another reason is that the proposed affinity-based algorithm uses the *interest* values instead of the *support* values in the first phase to determine the media object pairs in the candidate_pool. That is, not all the discovered association rules which pass the minimum support and minimum confidence factors are interesting enough to capture the quasi-equivalence relationships, as can be seen from the results of both empirical studies. Apparently, using the *interest* values can better indicate the usefulness of the rules in the discovery of the quasi-equivalent media object pairs. In addition, the *interest* constraint and the *refinement* constraint are used in the first phase to improve the performance in the proposed approach. The *interest* constraint is used to check the interestingness of the media object pairs to remove the unsatisfied media object pairs, and the *refinement* constraint is applied to allow more iterations to be executed to refine the results.

5. Conclusions

In this paper, we have proposed a generalized affinity-based association rule mining approach to discover the set of quasi-equivalent media objects from a network of databases. The quasi-equivalent relationship is used to approximate the structurally equivalent relationship. A new set of affinity-based measures to augment the standard measures of support, confidence, and interest is presented.

The affinity-based measures are both intuitively reasonable and understandable since they consider the access frequencies of queries and can be incorporated into current item set algorithms with no decrease in efficiency. The mining process is structured by a two-phase architecture that provides more informative feedback via conducting several user-specified constraint checkings.

We gave an algorithm for mining such affinity-based associations and conducted two empirical studies on the real database management systems. The results of the empirical studies show that the proposed approach not only detects the set of quasi-equivalent media objects which matches the structurally equivalent media object pairs known to be existing in the databases, but also performs better than the basic association rule mining approach in discovering the quasi-equivalence relationships. Clearly, discovering the quasi-equivalence relationships for media objects in a network of databases can assist in improving query processing performance. The more the databases there are, the more query processing performance improvement can be achieved.

Acknowledgements. This work has been partially supported by the National Science Foundation under contract IRI 9619812.

References

Agrawal R, Imielinski T, Swami A (1993) Mining association rules between sets of items in large databases. In Proceedings of 1993 ACM SIGMOD Conference on Management of Data, Washington DC, USA, pp 207–216

Candan KS, Rangan PV, Subrahmanian VS (1998) Collaborative multimedia systems: synthesis of media objects. IEEE Transactions on Knowledge and Data Engineering 10(3):433–457

Cheeseman P, Stutz J (1996) Bayesian classification (AutoClass): theory and results. In Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (eds). Advances in knowledge discovery and data mining. AAAI/MIT Press, Cambridge, MA, pp 153–180

Chen MS, Han J, Yu PS (1996) Data mining: an overview from a database perspective. IEEE Transactions on Knowledge and Data Engineering 8(6):866–883

Chen S-C, Kashyap RL (1997) Temporal and spatial semantic models for multimedia presentations. In 1997 international symposium on multimedia information processing, Taipei, Taiwan, pp 441–446

Chen S-C, Kashyap RL (1999) A spatio-temporal semantic model for multimedia presentations and multimedia database systems. IEEE Transactions on Knowledge and Data Engineering (accepted for publication)

Date CJ (1995) An introduction to database systems (6th edn). Addison-Wesley, Reading, MA

Elder IV JF, Pregibon D (1996) A statistical perspective on knowledge discovery in databases. In Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (eds). Advances in knowledge discovery and data mining. AAAI/MIT Press, Cambridge, MA, pp 83–113

Elmasri R, Navathe SB (1994) Fundamentals of database systems (2nd edn). Benjamin/Cummings, Redwood City, CA

Ester M, Kriegel HP, Xu X (1995) Knowledge discovery in large spatial databases: focusing techniques for efficient class identification. In Proceedings of the fourth international symposium in large spatial databases (SSD '95), Portland, Maine, USA, August 1995, pp 67–82

Fayyad UM, Piatetsky-Shapiro G, Smyth P (1996) From data mining to knowledge discovery: an overview. In Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (eds). Advances in knowledge discovery and data mining. AAAI/MIT Press, Cambridge, MA, pp 1–34

Inmon WH (1992) Building the data warehouse. QED Technical Publishing Group, Wellesley, MA

Langley P (1996) Elements of machine learning. Morgan Kaufmann, San Mateo, CA

Larson JA, Navathe SB, Elmasri R (1989) A theory of attribute equivalence in databases with application to schema integration. IEEE Transaction on Software Engineering 15(4):449–463

Lee H-Y, Ong H-L, Quek L-H (1995) Exploiting visualization in knowledge discovery. In Proceedings of the 1st international conference on knowledge discovery and data mining (KDD '95), Montreal, Canada, pp 198–203

Lu H, Setiono R, Liu H (1995) NeuroRule: a connectionist approach to data mining. In Proceedings of the 21st international conference on very large data bases, Zurich, Switzerland, September 1995, pp 478–489

Navathe SB, Elmasri R, Larson JA (1986) Integration user views in database design. IEEE Computer 19(January):50–62

Poe V (1996) Building a data warehouse for decision support. Prentice-Hall, Englewood Cliffs, NJ

Rosenthal A, Heiler S (1987) Querying part hierarchies: a knowledge-based approach. In Proceedings of the ACM/IEEE design automation conference, Miami Beach, FL, USA

Shavlik JW, Dietterich TG (eds) (1990) Readings in Machine Learning. Morgan Kaufmann, San Mateo, CA

Shyu M-L, Chen S-C, Kashyap RL (1998a) Database clustering and data warehousing. In Proceedings of the 1998 ICS workshop on software engineering and database systems, Tainan, Taiwan, 17–19 December 1998, pp 30–37

Shyu M-L, Chen S-C, Kashyap RL (1998b) Information retrieval using Markov model mediators in multimedia database systems. In Proceedings of the 1998 international symposium on multimedia information processing, Chung-Li, Taiwan, 14–16 December, pp 237–242

Shyu M-L, Chen S-C, Kashyap RL (1999) Discovering quasi-equivalence relationships from database systems. In Proceedings of the ACM eighth international conference on information and knowledge management (CIKM '99), Kansas City, MO, USA, 2–6 November 1999, pp 102–108

Simoudis E, Livezey B, Kerber R (1996) Integrating inductive and deductive reasoning for data mining. In Fayyad UM, Piatetsky-Shapiro G, Smyth P, Uthurusamy R (eds). Advances in knowledge discovery and data mining. AAAI/MIT Press, Cambridge, MA, pp 353–373

Srikant R, Agrawal R (1995) Mining generalized association rules. In Proceedings of the 21nd international conference on very large databases, Zurich, Switzerland, September 1995, pp 407–419

Srikant R, Agrawal R (1996) Mining quantitative association rules in large relational tables. In Proceedings of the 1996 ACM SIGMOD international conference on management of data, Montreal, Canada, June 1996, pp 1–12

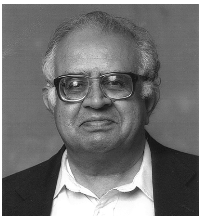
Su SY, Krishnamurthy V, Lam H (1989) An object oriented semantic association model (OSAM) for modeling CAD/CAM Databases. In Kumara S, Kashyap RL, Soyster AL (eds). Artificial intelligence: manufacturing theory and practice. American Institute of Industrial Engineers, Norcross, GA, pp 463–494

Zhang T, Ramakrishnan R, Livny M (1996) BIRCH: an efficient data clustering method for very large databases. In Proceedings of the 1996 ACM SIGMOD international conference on management of data, Montreal, Canada, June 1996, pp 103–114

Author Biographies

Mei-Ling Shyu received her PhD from the School of Electrical and Computer Engineering, Purdue University, West Lafayette, Indiana, USA in 1999. She also received her MS in Computer Science, MS in Electrical Engineering, and MS in Restaurant, Hotel, Institutional, and Tourism Management from Purdue University, West Lafayette, IN, USA in 1992, 1995, and 1997, respectively. She has been an Assistant Professor at the Department of Electrical and Computer Engineering, University of Miami since January 2000. Her research interests include data mining, data warehousing, information retrieval, digital library, multimedia database systems, multimedia information systems, object-oriented database systems, distributed database systems, and heterogeneous database systems. She is a member of the IEEE, IEEE Women in Engineering, ACM, and ACM SIGMOD.

Shu-Ching Chen received his PhD from the School of Electrical and Computer Engineering from Purdue University, West Lafayette, Indiana, USA in December 1998. He also received his Computer Science, Electrical Engineering, and Civil Engineering Master degrees from Purdue University, West Lafayette, IN. He has been an Assistant Professor in the School of Computer Science, Florida International University (FIU) since August, 1999. He has authored one book and more than 40 publications, including *IEEE Transactions on Knowledge and Data Engineering*, *VLDB*, *IEEE ICDE*, *IEEE ICME*, *ACM Multimedia*, *ACM GIS*, etc. His main research interests include distributed multimedia database systems and information systems, information retrieval, object-oriented database systems, data warehousing, data mining, and distributed computing environments for intelligent transportation systems (ITS). He was the program co-chair of the 2nd International Conference on Information Reuse and Integration (IRI-2000). He is a member of the IEEE Computer Society, ACM, and ITE.



R. L. Kashyap received his PhD in 1966 from Harvard University, Cambridge, Massachusetts. He joined the staff of Purdue University in 1966, where he is currently a Professor of Electrical and Computer Engineering and the Associate Director of the National Science Foundation supported Engineering Research Center Intelligent Manufacturing Systems at Purdue. He is currently working on research projects supported by the Office of Naval Research, Army Research Office. NSF, and several companies like Cummins Engines. He has directed more than 40 PhD dissertations at Purdue. He has authored two books and more than 300 publications, including 120 archival journal papers in areas such as pattern recognition, random field models, intelligent data bases, and intelligent manufacturing systems.

Correspondence and offprint requests to: Mei-Ling Shyu, Department of Electrical and Computer Engineering, University of Miami, Coral Gables, FL 33124-0640, USA.
Email: shyu@miami.edu