Discovering Quasi-Equivalence Relationships from Database Systems

Mei-Ling Shyu
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907
shyu@ecn.purdue.edu

Shu-Ching Chen
School of Computer Science
Florida International University
Miami, FL 33199
chens@cs.fiu.edu

R. L. Kashyap
School of Electrical and Computer Engineering
Purdue University
West Lafayette, IN 47907
kashyap@ecn.purdue.edu

Abstract

Association rule mining has recently attracted strong attention and proven to be a highly successful technique for extracting useful information from very large databases. In this paper, we explore a generalized affinity-based association mining which discovers quasi-equivalent media objects in a distributed information-providing environment consisting of a network of heterogeneous databases which could be relational databases, hierarchical databases, object-oriented databases, multimedia databases, etc. Online databases, consisting of millions of media objects, have been used in business management, government administration, scientific and engineering data management, and many other applications owing to the recent advances in high-speed communication networks and large-capacity storage devices. Because of the navigational characteristic, queries in such an information-providing environment tend to traverse equivalent media objects residing in different databases for the related data records. As the number of databases increases, query processing efficiency depends heavily on the capability to discover the equivalence relationships of the media objects from the network of databases. Theoretical terms along with an empirical study of real databases are presented.

1 Introduction

In the last decade, the exponential growth of computer networks and data-collection technology has generated an incredibly large offering of products and services for the users of computer networks. 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. However, our ability to interpret and analyze the data is still limited, creating an urgent need to accelerate discovery of information in databases. As pointed out by [9], there is a need and an opportunity for at least a partially-automated form of *knowledge discovery in databases (KDD)*, or *data mining* to handle the huge size of real-world database systems. In [5], the authors define *knowledge discovery in database (KDD)* to be the non-trivial process of identifying valid, novel, potentially useful, and ultimately understandable patterns in data and *data mining* to be the application of algorithms for extracting patterns from data. In other words, data mining is a component in the KDD process and concerns with the means by which patterns are extracted and enumerated from the data. However, in most of the existing articles and documents, the terms *knowledge discovery in databases*, *knowledge mining from databases*, *knowledge extraction*, *data archaeology*, *data dredging*, *data analysis*, etc. carry a similar or slightly different meaning [3]. Here, we use the terms KDD and data mining interchangeably.

In the previous study, we proposed a probabilistic network-based model, called *Markov model mediators* (MMMs), that allows us to query different media types and manage the rich semantic multimedia data for multimedia databases [11] [12]. With the help of probabilistic networks, methods can be developed to discover useful information and knowledge for the multimedia databases via probabilistic reasoning. Since the primitive constructed or manipulated entities in most multimedia systems are called *media objects* which could be a video clip, an image, a text file, or a complex entity of these simpler entities [2], a media object is represented

as a node in an MMM and is associated with an *augmented transition network (ATN)* which is a model for multimedia presentations, multimedia database searching, and multimedia browsing [4].

The proposed MMM mechanism facilitates the functionality of an MDBMS (*multimedia database management system*) by three steps. First, a stochastic process performs probabilistic reasoning to derive sets of probability distributions from a set of historical data and build a probabilistic network. The set of historical data is used as the training traces for finding the probability distributions. Second, a federation of data warehouses is constructed based on the mined probability distributions [10]. Third, a second stochastic process generates a list of possible state sequences with respect to a given query and indicates which particular media objects to query over the constructed data warehouses [11]. When the required media objects are predicted, the corresponding ATNs are traversed for information retrieval. Here, MDBMSs are considered since an MDBMS stores and manages not only images, audio, graphics, animation, and full-motion video, but also text as in traditional text-based databases. Also, data access and manipulation for multimedia databases are more complicated than those of the conventional databases since it is necessary to incorporate diverse media with diverse characteristics.

Because of the navigational characteristic, queries tend to access related data records from equivalent media objects which span multiple multimedia databases. In a single database, media object equivalence cannot exist since a database schema represents a non-redundant view. As such, only media objects across different databases can have an equivalence relationship. Two media objects are said to be equivalent if they are deemed to possess the same real world states *(RWS's)* [6] [7], i.e., if they represent the same sets of instances of the same real world entity. Experimental results in [10] [12] showed that the better the federation of data warehouses is, the more the cost of query processing is reduced. The construction of data warehouses requires that two databases can be clustered in the same data warehouse only if these two databases have some quasi-equivalent media objects. However, all the media object quasi-equivalence relationships were assumed given as the prior knowledge. To make the construction of data warehouses fully automatic without any given prior knowledge, the set of quasi-equivalent media objects needs to be explored.

In this paper, we explore a new data mining capability that involves mining quasi-equivalent media objects in a network of databases where queries tend to access information from related or quasi-equivalent media objects residing across multiple databases. We use relative affinity measures (as defined in [10]) 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 itemset algorithms with no decrease in efficiency. Clearly, discovering the quasi-equivalence relationships for media objects will not only help automate the construction of data warehouses but also lead to better query performance.

As the number of databases increases, query processing performance depends heavily on the capability to discover the equivalence relationships of the media objects from the network of databases. We implemented the algorithms and conducted an empirical study on the real database management systems at Purdue University to examine the performance of our proposed approach. The result shows that our fine algorithm can exploit the set of quasi-equivalent media objects correctly so that the previously unknown knowledge can be discovered.

This paper is organized as follows. In next section, we briefly introduce the association rules, the affinity-based association rules, and the definitions of affinity-based support, confidence, and interest measures. In Section 3, we propose our generalized affinity-based association mining approach. An empirical study is given in Section 4. Section 5 concludes the paper.

2 Affinity-Based Association Rules

2.1 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. 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.

Association discovery was first introduced by [1]. 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 contain X tend to contain Y. Each association rule is assigned a support factor and a confidence factor. The support factor indicates the relative occurrence of the detected association rules 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 degree to which the rule is true across individual records and is defined as the ratio of the number of tuples satisfying both X and Y over the number of tuples satisfying X. The problem is to find all the association rules satisfying user-specified minimum support and minimum confidence constraints that hold in a given database. An example of an association rule is: "80% of transactions contain bread also contain butter; 40% of transactions contain both bread and butter." Here, this association rule is supported by 40% of the database records and the confidence factor is 80%. Rules with high support and confidence factors represent a higher degree of relevance than rules with low support and confidence factors.

Notice that not all the discovered association rules which pass the minimum support and minimum confidence factors are interesting enough to present and sometimes they might be misleading [3]. Hence, an interest factor is defined to filter out such kind of misleading. However, the interestingness or the usefulness of a rule is often application-dependent. There have been several studies on quantifying the interestingness or usefulness of the discovered rules in the literature [8] [13].

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

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

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

$$confidence(X \rightarrow Y) = \frac{P(X \cap Y)}{P(X)} = \frac{\mid X \cup Y \mid}{\mid X \mid} \quad (3)$$

$$interest(X \rightarrow Y) = \frac{P(X \cap Y)}{P(X)P(Y)} \quad (4)$$

2.2 Relative Affinity Measures

We use the relative affinity values to measure how frequently two media objects have been accessed together in a set of queries [10]. 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. However, the current definition of *support* tells only the number of transactions containing an itemset but not the number of items. An item may be purchased in multiples in a transaction such that it should be considered more frequently than the support measure indicates. Similarly, each query could have a distinct frequency, i.e., a query may be activated several times. 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.

Let m and n be media objects, q the total number of queries, $access_k$ the access frequency of query q_k per time period, and $use_{m,k}$ the usage pattern with value 1 if media object m is accessed by query q_k and value 0 otherwise. $access_k$ and $use_{m,k}$ are available from the historical data. Define

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

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

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

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

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

Then, the generalized association mining is performed to determine the set of quasi-equivalent media objects. However, since we try to discover the quasi-equivalence relationship of two media objects, only the 2-itemsets are considered hence reducing the overheads such as database scans and large itemset generations.

3 Generalized Affinity-Based Association Mining

3.1 Architecture

Figure 1 shows the architecture for the proposed generalized affinity-based association mining. The multimedia resource subsystem consists of four modules – multimedia resources, multimedia resource schemas, resource databases, and a set of historical data. Each multimedia resource is associated with a designated resource schema which defines the set of media object definitions with their attributes. The resource database is a set of persistent objects which are instances of the media objects defined in the schema. The set of historical data includes the usage patterns of the media objects with respect to the set of sample queries and the access frequencies of the sample queries. The multimedia resource databases together with the set of historical data

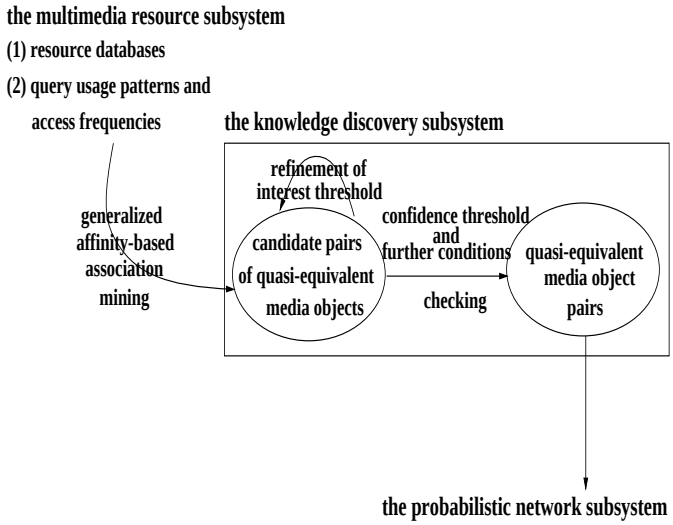

Figure 1: *Architecture for Generalized Affinity-Based Association Mining.*

provide the prior information of the knowledge discovery subsystem. The *generalized affinity-based association mining* process supports the knowledge discovery subsystem and the discovered knowledge (i.e., the pairs of quasi-equivalent media objects) is then used to assist in constructing the federation of data warehouses.

The association mining process consists of two phases. 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.

3.2 Algorithm

In this subsection, we describe the algorithm for discovering the set of quasi-equivalent media object pairs. Starts with all the media objects in the databases. Let L_1 and L_2 represent the sets of 1-itemsets and 2-itemsets, where each 1-itemset has one media object and each 2-itemset 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 summary of $access_k$ are computed. After that, $support(m \rightarrow n)$ and $interest(m \rightarrow n)$ 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. Assume the number of media objects in the databases is Nmo. The values for $cria1$, $cria2$, and $Conf$ need to be decided by the users before the algorithm is run. The variable $cria1$ sets the minimum interest threshold for each iteration. Let the maximal interest value for the media object m to be I_m and the resulting set to be candidate_pool. The minimum interest threshold is defined to be "iteration number $\times cria1 \times I_m$". In this case, the minimum interest threshold increases as the number of iterations increases. The variable $cria2$ sets the refinement constraint for Phase I. The refinement constraint threshold is defined to be "$cria2\times$ the total number of media objects". If the number of media objects which have zero or one pair remaining in the candidate_pool is greater than or equal to the refinement constraint, then Phase I stops and goes to Phase II. Otherwise, go to next iteration with a new minimum interest threshold for Phase I. The variable $Conf$ sets the minimum confidence threshold for Phase II. This value is used to remove the pairs which fail the minimum confidence threshold checking.

⋆ **Steps** for Phase I:

1. For all the 1-itemsets, compute $support(m)$ (Equation 6).
2. For all the 2-itemsets,
   - 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 from $interest(m \rightarrow n)$ 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.

(b) Set the minimum interest threshold $IntTd = cria1 \times iter \times I_m$.

(c) For those media objects n's,
if $iter = 1$ and $interest(m \rightarrow n) \geq IntTd$,
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 $\geq cria2 \times Nmo$,
then goto Phase II.
else set $iter = iter + 1$ and goto step 4.

⋆ **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.

4 Empirical Study

To empirically test the proposed generalized affinity-based association mining approach, we ran the algorithm on the financial database management systems at Purdue University in July, August, and September for the year 1997. We implemented the algorithm with the affinity-based support, confidence, and interest measures reflecting the number of accesses for each media object. 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. Set $cria1 = 0.2$, $cria2 = 0.5$, and $Conf = 99\%$. Table 1 lists the maximal interest values for the *22* media objects.

Let the media objects be numbered from 1 to 22 and the media objects in the same database have consecutive numbers. Set $cria1 = 20\%$, $cria2 = 50\%$, and

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

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

$Conf = 99\%$. 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 on 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. 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 the I_m and I_n are equal, then m and n are potentially to be 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.

When the I_m measures are determined, the $IntTd$ for the first iteration is set to be $0.2 \times I_m$ 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). 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. 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 confidence threshold $Conf$ is set to be 99%. There are 24 pairs left in the candidate-pool after the confidence constraint checking. 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 in media objects in 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. Since the quasi-equivalence relationship (m, n) is the same as the quasi-equivalence relationship (n, m), there are eight quasi-equivalent pairs when the order is not considered.

5 Conclusions

In this paper, we proposed a generalized affinity-based association mining approach to discover the set of quasi-equivalent media objects from a network of heterogeneous databases in a distributed information-providing environment. The quasi-equivalent relationship is used to approximate the structurally equivalent relationship. We have presented a new set of affinity-based measures to augment the standard measures of support, confidence, and interest. Affinity-based measures are both intuitively reasonable and understandable since they consider the access frequencies of queries and can be incorporated into current itemset algorithms with no decrease in efficiency. The mining process is structured using a two-phase architecture which provides more informative feedback via conducting several user-specified constraint checkings.

We gave an algorithm for mining such affinity-based associations and the quasi-equivalent relationship is used to approximate the structurally equivalent relationship. The results of our empirical study on the real database management systems show that the proposed approach detects the set of quasi-equivalent media objects which matches the structurally equivalent media object pairs known to be existing in the databases. Clearly, discovering the structural equivalence relationships for media objects will not only help automate the construction of data warehouses but also lead to better query performance.

6 Acknowledgements

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

References

[1] R. Agrawal, T. Imielinski, A. Swami, "Mining association rules between sets of items in large databases, " Proc. 1993 ACM SIGMOD Conference on Management of Data, pp. 207-216, 1993.

[2] K.S. Candan, P.V. Rangan, and V.S. Subrahmanian, "Collaborative multimedia systems: synthesis of media objects," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 10, No. 3, pp. 433-457, May/June 1998.

[3] M.S. Chen, J. Han, and P.S. Yu, "Data mining: An overview from a database perspective," *IEEE Transactions on Knowledge and Data Engineering*, pp. 866-883, Vol. 8, No. 6, December 1996.

[4] S-C. Chen and R.L. Kashyap, "A spatio-temporal semantic model for multimedia presentations and multimedia database systems," accepted for publication on *IEEE Transactions on Knowledge and Data Engineering*, 1999.

[5] U.M. Fayyad, G. Piatetsky-Shapiro, and P. Smyth, "From data mining to knowledge discovery: An overview," in U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, editors, *Advances in Knowledge Discovery and Data Mining*, pp. 1-34, AAAI/MIT Press, 1996.

[6] J.A. Larson, S.B. Navathe, and R. Elmasri, "A theory of attribute equivalence in databases with application to schema integration," *IEEE Transaction on Software Engineering*, vol. 15, no. 4, Apr. 1989.

[7] S.B. Navathe, R. Elmasri, and J.A. Larson, "Integration user views in database design," *Comput.*, vol. 19, Jan. 1986.

[8] G. Piatetsky-Shapiro and C.J. Matheus, "The interestingness of deviations," presented at the AAAI Workshop on Knowledge Discovery in Databases, Seattle, 1994.

[9] G. Piatetsky-Shapiro, "Knowledge discovery in real databases: A report on the IJCAI-89 Workshop,"

AI Magazine, vol. 11, no. 5, Special issue, pp. 69-70, Jan. 1991.

[10] M-L. Shyu, S-C. Chen, and R. L. Kashyap, "Database Clustering and Data Warehousing," 1998 ICS Workshop on Software Engineering and Database Systems, pp. 30-27, Dec. 17-19, 1998.

[11] M-L. Shyu, S-C. Chen, and R. L. Kashyap, "Information Retrieval Using Markov Model Mediators in Multimedia Database Systems," 1998 International Symposium on Multimedia Information Processing, pp. 237-242, Dec. 14-16, 1998.

[12] M-L. Shyu and S-C. Chen, "Probabilistic Networks for Data Warehouses and Multimedia Information Systems," submitted to *IEEE Trans. on Knowledge and Data Engineering.*

[13] A. Silberschatz and A. Tuzhilin, "On subjective measure of interestingness in knowledge discovery," Proc. First Int'l Conf. Knowledge Discovery and Data Mining (KDD'95), pp. 275-281, Montreal, Canada, August 1995.