ISSN:1991-8178

Australian Journal of Basic and Applied Sciences

Journal home page: www.ajbasweb.com

A Method For Clustering Items In Multi Data Base Systems

[1]Mrs.R.Suganthi and [2]Dr.P.Kamalakannan

[1]*Department of Computer Applications / Bharadhidasan University, India*
[2]*Department of Computer Science, Periyar University, India*

ARTICLE INFO

Article history:
Received 3 October 2015
Accepted 31 October 2015

Keywords:
Clustering, Measure of Association, Multi data base mining, Patterns, Synthesis of pattern,

ABSTRACT

Nowadays, many organization is handling their branches across the world. Large companies managing a lot of information and taking decision based on the branch database along with the frequent patterns. Thus, it is essential to study data mining on multiple data bases. Here the major work in multiple database is finding the needed patterns for taking effective decision among all the patterns in the organization. This will leads to company growth and describes the individuality of the few branches. Therefore, it is interesting to identify such types of patterns. There are two important patterns namely global patterns and local patterns which is used to provide the exact details of the company across all branch data bases. In this paper, We Propose frequent item sets in multiple databases by the clustering technique. For that, we are in need of two significant steps, first is measure of association and the second is synthesizing support of an item set. Experimental results are presented on both actual and artificial databases. We compare the proposed algorithm with the existing algorithm theoretically as well as experimentally. The experimental results show that the proposed algorithm is effective and promising.

To Cite This Article: Mrs.R.Suganthi and Dr.P.Kamalakannan., A Method For Clustering Items In Multi Data Base Systems. ***Aust. J. Basic & Appl. Sci.,*** *9(33): 115-121, 2015*

INTRODUCTION

The numeral branches of a multi-national company as well as the number of multi-national companies is growing over time due to a moderate cost-effective strategy espouse by many countries crosswise the world. Moreover, the economies of many countries are mounting at a quicker rate. As a result the number of multi-branch companies within a country is also increasing. Many of these companies collect a huge amount of data through different branches. Consider a multi-branch company that transacts from multiple branches. Each branch maintains a separate database for the transactions made at the branch. Thus, the company deals with multiple transactional databases. Data mining and knowledge discovery from outsized database is frequently measured as the source of many decision-support applications. Global decisions could be taken efficiently using such kind of patterns. Thus, it is important to associate local frequency item set in multiple databases. An common idea of the existing measures of association is presented here (Animesh Adhikari, 2013). For the purpose of selecting the suitable technique of mining multiple databases, we have surveyed the existing multi-database mining techniques. A study on the related clustering techniques (Chen, L., 2012) is also covered here. The notion of high frequency item sets is introduced (Adhikari, A., P.R. Rao, 2008) and an algorithm for synthesizing supports of such item sets is considered. The existing clustering technique might cluster local frequency items at a low level, since it estimates association among items in an item set with a low performance, and the designed new algorithm for clustering local frequency items is proposed here. An item set is a collection of items in a database. Each item set in a database is associated with a statistical measure called support (Hershberger, S.L., D.G. Fisher, 2005). Support of an item set X in database D is the fraction of transactions in D containing X, denoted by S(X, D). In general, let S(E, D) be the support of a Boolean expression E defined on the transactions in database D. An itemset X is called frequent in D if S(X, a, where a is user defined level of minimum support. If X is frequent then Y c X is also frequent, since S(Y, D) S(X, D), for Y # 0. Thus, each item of a frequent itemset is also frequent. Itemset could be considered as a basic type of pattern in a transactional database. Frequent patterns can be classified in various ways based on completeness of patterns, level of abstraction, number of data dimension and kinds of rules with patterns to be

Corresponding Author: Mrs.R.Suganthi, Department of Computer Applications / Bharadhidasan University, India
E-mail: mcasuganthi2011@gmail.com

Australian Journal of Basic and Applied Sciences, 9(33) October 2015, Pages: 115-121

mined. The collection of frequent item sets determines major quality of all the databases.

II. Related Work:

Multi data base mining has been documented as an important research topic in data mining (Adhikari, A., 2010; Zhang, S., 2004) Local pattern analysis has been proposed for Multi data base mining by Zhang et al and Later, absolute model of local pattern analysis is also projected by him. This absolute model helps to mine and analyse multiple large databases approximately. To improve the accuracy of multi data base mining, Zhang introduced the pipeline feedback technique. The item set pattern based clustering technique is used in multiple databases. Lin et al used two steps. As a first step, he proposed UMMI algorithm by using maximal item set property and potential item sets numbers has been reduced significantly. In second step, to determine all of the high utility item sets, UMMI used an effective lexicographic tree structure. Item set patterns based clustering procedure was proposed in multiple databases. Mining multiple databases dealt by Distributed data mining (DDM) algorithms and it is distributed over different geographical regions. Researchers have started addressing problems In the last few years, where the databases stored at different places cannot be moved to a central storage area for a variety of reasons. Different distributed sources of computation is often used for distributed data mining environment. The centralization of data is either not possible, or at least not always desirable for advent of ubiquitous computing (Greenfield, A., 2006), sensor networks (Zhao, F. and L. Guibas, 2004), grid computing (Wilkinson, B., 2009), and privacy-sensitive multiparty data (Kargupta, H., 2008). Jaroszewicz and Simovici (Jaroszewicz, S., D.A. Simovici, 2002) proposed a method for estimating supports of frequent item sets using Bonferroni-type inequalities (Galambos, J. and I. Simonelli, 1996) In the context of support estimation of frequent itemsets. Also, Pavlov et al (2000) proposed the maximum-entropy approach to sup-port estimation of a general Boolean expression.. Where problems that deal with single database, these support estimation techniques might be suitable. To deal with multiple databases, existing parallel mining techniques (Agrawal, R. and J. Shafer, 1999; Chattratichat, J., 1997; Cheung, D., 1996) could also be used. Parallel algorithms provide a costly solution to multi-databases mining and it is also designed with a different objective. Cost benefit analysis should be made before implementing such a decision, since it might require high investment on hardware and software. It might not be an acceptable solution by the company management in many situations,. Moreover, when a traditional data mining technique is applied to the entire dataset, it might be difficult to find regional patterns. To treat with multiple databases existing parallel mining techniques could also be used. For mining item set patterns in multiple databases, technique is needed in this context,. Afterwards, using a measure of association, association among items in an item set is captured. Finally, to cluster local frequency items in multiple databases a clustering algorithm is designed. Therefore, three major areas have been categorized for the proposed work viz., measures of association, synthesizing support of an item set, and clustering the frequent item set.

III. Measures of association:

To examine, relationship among items in multiple databases is the important role in data mining problems. Among a set of items, Measure of association gives a numerical estimate of statistical dependence. An overview of twenty one interestingness measures in the statistics, machine learning and data mining literature have presented by Tan et al (2002) in the context of similarity measures. Between two sets of items in large databases, the item set measures, i.e ., support and confidence are used to identify frequent items in association rules.

Our first measures sim1 to measure jacord measures such us support ,interest,cosine do not serve us good measures of similarity since the denominators are not relevant. Support measure is proposed by Agarwal et al (1993) in the context of finding association rules in a database. Ramkumar et al. (2013) reviewed the multi-database mining recent research contributions which provides the detailed information of various author's description. Here the number of frequency items in database is important to find the support of an items set, in which item set in a transactions is a source of items. But, support of an item set does not consider frequencies of its subsets. As a result, the support of an item set might not be a good measure of association among items in an item set. In recent multi data base mining papers, various measures of association is explained .In some circumstance, some existing measures are not able to detain the association among a set of items in a data base accurately. There are two generalized measure of association used namely A1 and A2. We can find the relationship among the databases like D1,D2 easily using the method of Similr2 generalized measure association. In this paper, measure A2 provides the information about clustering the frequent items in multiple data bases.

IV. Support of an item set:

local pattern analysis in multi database mining could be viewed as a two step process.

- Mining of the each local data base
- Synthesizing the patterns

To improve the quality of global patterns and mining multiple large databases ,pipe lined feedback technique (PFT) has been used.

Australian Journal of Basic and Applied Sciences, 9(33) October 2015, Pages: 115-121

Consider a multi-branch company that has four branches. Let Di be the database corresponding to the i-th branch, for i = 1, 2, 3, 4. The branch databases are given as follows. DI = {{a, b}, {a, b, c}, {a, b, c, d}, {c, d, e}, {c, d, f}, {c, d, i}}; D2 — { {a, b}, {a, b, g}, {g}}; D3= {{a, b, d}, {a, c, d}, {c, 6}1, D4= {{a}, {a, b, c}, {c, d}, {c, d, i}l. Assume that a = 0.4, and y= 0.6. Let X(77) denotes the fact that the item set X has support value in the corresponding database. We sort databases in non-increasing order on database size (in bytes). The sorted databases are given as follows: DI, D4, D3, D2.

Applying PFM, the item sets in different local databases are given as follows:
LPB(D1, a) = {{a}(0.5), {b}(0.5), {c}(0.833), {d}(0.667), {a, b}(0.5), {c, d}(0.667)},
LPB(D4, a) = {{a}(0.667), {b}(0.25), {c}(0.75), {d}(0.25), {a, b}(0.333), {c,d}(0.667)},
LPB(D3, a) = {{a}(0.667), {b}(0.333), {c}(0.667), {d}(1.0), {a, b}(0.333),{c, d}(0.667)},
LPB(D2,a) = {{a}(0.667), {b}(0.667), {c}(0.0), {d}(0.0), {a,b}(0.667), {c, d}(0.0), {g}(0.667)}.
Let D = Synthesized HEISs in D are given as follows:
SHEIS(D, 0.4, 0.6) ={a}(0.563), {b}(0.438), {c}(0.563), {d}(0.563), {a, b}(0.438), {c, d}(0.5)}.

The collection of SHEISs of size greater than 1 forms the basis of the proposed clustering technique. We present below an algorithm to obtain synthesized association among items in each SHEIS of size greater than 1. Let N be the number of item sets in n databases. Let AIS be a two dimensional array such that AIS(i) is the array of item sets extracted from D for i = 1, 2, ..., n. Also, let IS be the set of all item sets in n databases. An item set could be described by the following attributes: item set, supp, and did. Here, item set, supp and did represent the item set, support and database identification of item set, respectively. All the synthesized item sets are kept in the array SIS. Each synthesized item sets has the following attributes: item set, SS, and SA. Here, SS and SA represent synthesized support and synthesized association of the item set, respectively and it is possible to extract various patterns through the synthesizing algorithms in which the significance of the patterns is different which is used in organization growth. In the following algorithm, we synthesize the association among items of each SHEIS.

V. Algorithm:

procedure Synthesize Association (n, AIS, size,)

Algorithm- Synthesizing the association among items:

Input:

ni: number of databases
AIS: Array of item sets extracted
Size: Number of transactions in input databases
δ: Threshold for minimum number of extractions of an item set
NSFIS : Number of Synthesized frequent item sets.

Output:

Synthesized association among items of each SHFIS

1. nsis=0;i=1;
2. while(j<i+n) do
3. if(is<j).itemset=IS(i).itemset) then
4. process IS(i);
5. end if
6. end while
7. synsupp=supps (IS(i).itemset ,D)
8. initialize Synthesized association to S
9. if(SIS(nsynIS).ITEMSET>2) THEN
10. SIS(nsynIS).sa=simlr2(nsynIS1.itemset,D)
11. end if
12. end procedure

VI. Clustering of Frequent items:

Local frequency items are the basic components of many important patterns in multiple databases. The main objective of this paper is to cluster local frequency items. The existing techniques of clustering multiple databases work as follows. A measure of similarity between two databases is proposed. Let there be m databases to be clustered. Then the similarities for mC2 pairs of databases are computed. An algorithm is designed based on the measure of similarity. Then based on a level of similarity, the databases are clustered into different classes. For the purpose of clustering databases, Wu et al. (2005) have proposed the following measure of similarity between two databases.

Here I(Di) is the set of items in the database Di, i = 1, 2. Thus, sim1 is the ratio of number of items common to databases and the total number of items in the two databases. Measure sim1 could also be used to find similarity between two items in a database. One could derive measure sim1 from A2 . In other words, the measure sim1 is a special case of measure A2. Measure A1 (Adhikari, A., P.R. Rao, 2008) can also be used to find similarity among items in a database. In the following example, we show that association among items of an item set could not be determined correctly using associations of all possible subsets of size 2. In particular, association among items of {a, b, c} could not be correctly estimated by the associations between the items in {a, b}, {a, c}, and {b, c}. This issue has been explained in Example 1.

Example 1:

Let D5 = {{a, b, c, d}, {a, b, c, e}, {a, b, d}, {a, e, f}, {b, c, e}, {d, e, g}, {d, f, g}, {e, f, g}, {e, f, h}, {g, h, i}}. Also, let a be 0.2. The supports of relevant frequent itemsets are given as follows. S({a}, D5) = 0.4, S({b}, D5) = 0.4, S({c}, D5) = 0.3, S({a, b}, D5) = 0.3, S({a, c}, D5) = 0.2, S({b, c}, D5) = 0.3, S({a, b, c}, D5) = 0.2. Now, sim1({a, b}, D5) = 0.6,

sim1({a, c}, D5) = 0.4, sim1 ({b, c}, D5) = 0.75. Using sim1, the items a, b, and c could be put in the same class at the level of similarity 0.4, i.e., minimum{0.6, 0.4, 0.75}. Using A2, we have A2({a, b, c}, D5) = 0.67. Thus, the items a, b, and c could be put in the same class at the level 0.62. The subset of transactions {{a, b, c, d}, {a, b, c, e}, {a, b, d}, {a, e, f}, {b, c, e}} of D5 results in the amount of association among a, b, and c. Three out of five transactions contain at least two items of {a, b, c}. Two out of five transactions contain all the items of {a, b, c}. More the number of items of {a, b, c} occur simultaneously, higher is the association among items of {a, b, c}. Thus, we examine that the amount of association among the items of {a, b, c} is close to 0.62 rather than 0.4. Thus, one might fail in measuring association correctly among the items of {a, b, c} based on the comparison between items of {a, b}, {a, c}, and {b, c}.

VII. Experimental result:

We have carried out several experiments to study the effectiveness of our approach. All the experiments have been implemented on a 2.8 GHz Pentium D dual processor with 512 MB of memory using Net Beans IDE 6.0 software. We present the experimental results using three real databases. The database retail is obtained from an anonymous Belgian retail supermarket store. The database mushroom is also available and the database ecoli has been processed for the intention of conducting experiments. For this purpose, we have omitted non-numeric fields of all the mentioned database. We present some characteristics of these databases in Table 1.

Table 1: Database characteristics.

Dataset	NT	ALT	AFI	NI
Check	400	3.03	3.1007	39
Retail	8000	7.9067	21.8769	106
SD100000(S)	5000	11.66	12.22	100
Ecoli	336	7.00	25.65	92

Let NT, ALT, AFI, and NI denote the number of transactions, average length of a transaction, average frequency of an item and number of items in the data source respectively. Each of the above databases is divided into 3 databases for the purpose of conducting experiments. The databases obtained from, Check(C), Retail(R),SD100000 (S) and ecoli(E) are named as C,R, S and E respectively, for i = 0, 1, ..., 9. The databases Ci , R i , Si and Ei , are called input databases, for i = 0, 1, ..., 9. Some characteristics of these input databases are presented in Table 1. We have carried out several experiments to study the effectiveness of our approach for clustering the frequent items. We present results of the experiments based on the above databases. We observe that a partition of frequent items might not exist for some combination of input databases.

VIII. Overall output:

We sorted the item sets of IS, Accordingly that processing of item sets becomes easier. We find total number of transactions in different variables into variable 'ni'. The variables NSIS(n synthesized item sets) and *i* keep track of the number synthesized item sets and the current item sets of IS, respectively. An item sets gets processed at each iteration. An item set gets processed at each iteration. An item set occurs maximum *n* times. The variable count keeps track of number of times an item set is extracted. If an item set is highly extracted then we store the details into array SIS and increase NSIS by 1. We update the variable *i* by *j* for processing the next item set. likewise we processing the next item set then the next step is synthesized association among the items. Finally we could determine the time complexity of the algorithm. after run ,The databases are,

D1,D2,D3,D4,D5,D6,D7,D9,D8,D10

item1 == [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 18,2]

a) For experiment with Check(C): The set of frequent items in different databases are given as follows: FI(0, 9, 0.1) = {0}, {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}, {9}, {32}, {38},{39}, {48}). SHEISs of size greater than 1 along with their synthesized associations are given as follows: {39, 48} (0.443690), {39, 41, 48} (0.393977), {39, 41} (0.263936), {41, 48} (0.251072), {38, 39} (0.181348).

(b) For experiment with Retail(R): The set of frequent items in different databases are given as follows : FI(0, 9, 0.5) = {{1}, {2}, {3}, {6}, {7}, {9}, {10}, {11}, {23}, {24}, {28}, {29}, {34}, {36}, {37}, {38}, {39}, {48}, {52}, {53}, {54}, {56}, {58 }, {59}, {61}, {63}, {66}, {67}, {76}, {85}, {86}, {90}, {93}, {94}, {95}, {99}, {101}, {102 }, {110}, {114}, {116}, {117}, {119} }. Top 10 SHEISs of size greater than 1 along with their synthesized associations are given as follows: {34, 90} (0.999957), {34, 86} (0.999458), {34, 85} (0.995638), {34, 36, 85} (0.989711), {34, 36, 90} (0.987932), {34, 85, 90} (0.980130), {34, 36, 86} (0.977787), {34, 86, 90) (0.977439), {34, 85, 86) (0.968257), {85, 86) (0.962741).

Table 2: Input database characteristics.

DB	NT	ALT	AFI	NI	DB	NT	ALT	AFI	NI
C0	133	6.89	12.07	290	S0	1666	42.89	295.27	909
C1	133	7.67	12.25	234	S1	1666	41.99	286.27	968
C2	133	7.22	12.88	265	S2	1666	42.54	243.27	974
R0	2666	11.11	16.89	5600	E0	112	6.86	4.80	13
R1	2666	11.25	17.65	5678	E1	112	6.56	5.89	19
R2	2666	11.14	17.10	5571	E2	112	6.43	3.39	16

Table 3: Database Average Errors with synthesizing time & Clustering time.

Database	Average Error	SynTime(seconds)	Cls time(seconds)
D1	0.0876	5	23
D2	0.0932	9	6
D3	0.0422	10	7
D4	0.0405	6	1
D5	0.0379	5	31
D6	0.0344	6	25
D7	0.0582	9	1
D8	0.0623	10	16
D9	0.0522	11	27
D10	0.0661	12	29

IX. Comparing with the existing technique:

The proposed clustering technique is likely to enhance the accuracy of clustering process, if the clustering is performed at a level 6 such that 6 is a synthesized association of a class containing more than 2 frequent items. In each of the Tables 1,2 and 3, we present an example of clustering that achieves higher level of accuracy which is

Table 4: A Sample Clustering of frequent items in multiple databases.

α	β	Clustering	δ(existing approach)	δ (proposed approach)
0.1	0.7	{{0}, {1}, {2}, {3}, {4}, {5}, {6}, {7}, {8}, {9}, {32}, {38, 39}, {39, 41, 48}	0.400320	0.72841
0.5	0.7	{{l}, {2}, {3}, {6}, {7), {9), {10), {11}, {22}, {23}, {27}, {29), {36}, {33}, {38}, {47}, {51}, {52}, {53}, {55}, {57 }, {58), {20}, {23}, {24), {25}, {26}, {27}, {28), {29}, {30), {31), {32}, {33), {34), {35}, {36}, {38), {37}, {40), {41}, {42}, {43}, {44}, {45), {46}, {47}, {49}, {51}, {52}, {54}}	0.0661811	0.06961
0.3	0.7	{{0), {20), {23), {24), {25}, {26}, {27}, {28}, {29}, {30), {31), {32}, {33), {34), {35}, {36}, {38), {39}, {40),{41}, {42}, {43}, {44}, {47}, {48}, {51}, {58}, {59}}	0.0178571	0.232143

Table 5: Database Average Errors with synthesizing time.

α	β	Average error	Synthesizing time
0.5	0.4	0.07874	29
0.5	0.5	0.27415	22
0.5	0.6	0.87320	14
0.5	0.7	0.661811	11
0.5	0.8	0.185925	9
0.5	0.9	0.102765	8
0.5	0.1	0.562500	8

X. Synthesizing, Clustering time & Average error versus number of databases:

We have also studied the behavior of synthesizing and clustering time required over the number of databases used in this experiment. As the number of databases increases, the number of frequent item sets also increases. In general, we find that clustering time, synthesizing time and average error increases as the number of databases increases. In Figures 1,2 and 3 we present graphs of clustering time with average error for the experiments.

Fig. 1: Average Errors versus α at y= 0.7 (retail).

Conclusion:

The important task in many decision support system is clustering the relevant objects. Compare to the existing technique, our proposed new technique of clustering frequent items accuracy is higher degree. In the previous method, the main problem is that it might not be able to calculate accurate similar items with high frequency. The experimental results show that the proposed clustering method is efficient and shows potential patterns to the organization.

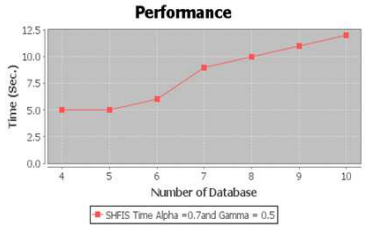

Fig. 2: Average Errors versus α at y= 0.7 (retail).

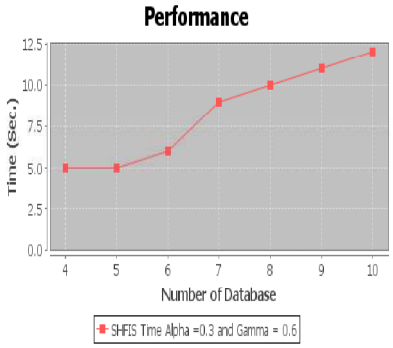

Fig. 3: Average Error versus α at y=0.7(check).

REFERENCES

Animesh Adhikari, 2013. Clustering local frequency items in multiple databases, Information sciences, 237: 221-241.

Chen, L., L. Zou, L. Tu, 2012. A clustering algorithm for multiple data streams based on spectral component similarity, Information Sciences, 183(1): 35– 47.

Adhikari, A., P.R. Rao, 2008. Synthesizing heavy association rules from different real data sources, Pattern Recognition Letters, 29(1): 59–71.

Hershberger, S.L., D.G. Fisher, 2005. Measures of Association, Encyclopedia of Statistics in Behavioral Science, John Wiley & Sons.

Adhikari, A., P. Ramachandrarao and W. Pedrycz, 2010. Developing Multi-Database Min- ing Applications, Springer.

Zhang, S., C. Zhang and X. Wu, 2004. Knowledge Discovery in Multiple Databases, Springer.

Zhang, S., C. Zhang and J.X. Yu, 2004. An efficient strategy for mining exceptions in multi-databases, Inform. Sciences, 165: 1–20.

Adhikari, A. and P.R. Rao, 2008. Synthesizing heavy association rules from different real data sources, Pattern Recognit. Lett., 29: 59–71.

Zhang, C., M. Liu, W. Nie and S. Zhang, 2007. Identifying global exceptional patterns in multi-database mining, IEEE Computat. Intell. Bull., 3: 19–24.

Lin, M.Y., T.F. Tu, S.C. Hsueh, 2012. High utility pattern mining using the maximal itemset property and lexicographic tree structures, Information Sciences, 215: 1–14.

Greenfield, A., 2006. Everyware: The Dawning Age of Ubiquitous Computing, New Riders Publishing.

Zhao, F. and L. Guibas, 2004. Wireless Sensor Networks: An Information Processing Approach, Morgan Kaufmann.

Wilkinson, B., 2009. Grid Computing: Techniques and Applications, CRC Press.

Kargupta, H., J. Han, P.S. Yu, R. Motwani and V. Kumar, 2008. Next Generation of Data Mining, Springer.

Jaroszewicz, S., D.A. Simovici, 2002. Support approximations using Bonferroni-type inequalities, in: Proceedings of sixth European Conference on Principles of Data Mining and Knowledge Discovery, 212–223.

Galambos, J. and I. Simonelli, 1996. Bonferroni-Type Inequalities with Applications, Springer.

Pavlov, D., H. Mannila and P. Smyth, 2000. Probabilistic models for query approximation with large sparse binary data sets, in: Proceedings of Sixteenth Conference on Un- certainty in Artificial Intelligence, 465–472.

Agrawal, R. and J. Shafer, 1999. Parallel mining of association rules, IEEE Trans. Knowl. Data Eng., 8: 962–969.

Chattratichat, J., J. Darlington, M. Ghanem, Y. Guo, H. Hüning, M. Köhler, J. Su- tiwaraphun, H.W. To and D. Yang, 1997. Large scale data mining: Challenges, and re- sponses, in: Proceedings of the Third International Conference on Knowledge Dis- covery and Data Mining, 143–146.

Cheung, D., V. Ng, A. Fu and Y. Fu, 1996. Efficient mining of association rules in dis- tributed databases, IEEE Trans. Knowl. Data Eng., 8: 911–922.

Tan, P.N. V. Kumar, J. Srivastava, 2002. Selecting the right interesting measure for associationpatterns, Proceedings of SIGKDD Conference, 32-41.

Agrawal, R., T. Imielinski, 1993. A. Swami, Mining association rules between sets of items in large databases, in: Proceedings of SIGMOD Conference on Management of Data, 207–216.

Ramkumar, T., S. Hariharan, S. Selvamuthukumaran, 2013. A survey on mining multiple data sources, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery, 3(1): 1–11.

Adhikari, A., P.R. Rao, 2008. Efficient clustering of databases induced by local patterns, Decision Support Systems, 44(4): 925–943.

Wu, X., C. Zhang, S. Zhang, 2005. Database classification for multi-database mining, Information Systems, 30(1): 71–88.

Adhikari, A., P.R. Rao, 2008. Capturing association among items in a database, Data & Knowledge Engineering, 67(3): 430–443.

Frequent Itemset Mining Dataset Repository. <http://fimi.cs.helsinki.fi/data/>.