

Association Rule Mining based on Ontological Relational Weights

N. Radhika, K.Vidya

Department of Computer Science and Engineering
Aurora's Technological and Research Institute, India

Abstract- Data mining has emerged to address the problem of transforming data into useful knowledge. On account of enormous of rules that can be produced by data mining algorithms, knowledge validation is one of the most problematic steps in an association rule discovery process. In this paper we propose a new WARM (weighted association rule mining) approach to prune and filter discovered rules based on the Ontological relational weights. We propose to use ontologies in order to improve the integration of user knowledge. An interesting real-life example and experimental results on different types of data are given; to reduce the numbers of rules to several dozen are less.

Index Terms- data mining, post processing, ontology, ORWM algorithm.

I. INTRODUCTION

ASSOCIATION rule mining, introduced in [1], is considered as one of the most important tasks in Knowledge Discovery in Databases [2]. Among sets of items in transaction databases, it aims at discovering implicative tendencies that can be valuable information for the decision-maker. An association rule is defined as the implication $X \rightarrow Y$, described by two interestingness measures support and confidence where X and Y are the sets of items and $X \cap Y = \emptyset$. Apriori [1] is the first algorithm proposed in the association rule mining field and many other algorithms were derived from it. Starting from a database, it proposes to extract all association rules satisfying minimum thresholds of support and confidence.

Unfortunately, the lower the support is, the larger the volume of rules becomes, making it intractable for a decision-maker to analyze the mining result. Experiments show that rules become almost impossible to use when the number of rules overpasses 100. Thus, it is crucial to help the decision-maker with an efficient technique for reducing the number of rules.

To overcome this drawback, several methods were proposed in the literature. On the one hand, different algorithms were introduced to reduce the number of itemsets by generating closed [3], maximal [4] or optimal itemsets [5], and several algorithms to reduce the number of rules, using nonredundant rules [6], [7], or pruning techniques [8]. On the other hand, postprocessing methods can improve the selection of discovered rules. Different complementary postprocessing methods may be used, like pruning, summarizing, grouping, or visualization [9]. Pruning consists in removing uninteresting or redundant rules. In summarizing, concise sets of rules are generated. Groups of rules are produced in the grouping process; and the visualization

improves the readability of a large number of rules by using adapted graphical representations. However, most of the existing post processing methods are generally based on statistical information in the database. Since rule interestingness strongly depends on user knowledge and goals, these methods do not guarantee that interesting rules will be extracted.

The classical model of association rule mining employs the support measure, which treats every transaction equally. In contrast, different transactions have different weights in real-life data sets. For example, in the market basket data, each transaction is recorded with some profit. Much effort has been dedicated to association rule mining with pre assigned weights. However, most data types do not come with such pre assigned weights, such as Web site click-stream data. There should be some notion of importance in those data. For instance, transactions with a large amount of items should be considered more important than transactions with only one item. Current methods, though, are not able to estimate this type of importance and adjust the mining results by emphasizing the important transactions. Apriori[1] is the first algorithm proposed in the association rule mining field and many other algorithms were derived from it. Unfortunately, the lower the support is, the larger the volume of rules becomes, making it intractable for a decision-maker to analyze the mining results.

In this paper, we introduce w-support, a new measure of item sets in databases with only binary attributes. The basic idea behind w-support is that a frequent item set may not be as important as it appears, because the weights of transactions are different. These weights are completely derived from the internal structure of the database based on the assumption that good transactions consist of good items. This assumption is exploited by extending Kleinberg's HITS[11] model and algorithm to bipartite graphs. Therefore, w support is distinct from weighted support in weighted association rule mining (WARM).where item weights are assigned. Furthermore, a new measurement framework of association rules based on w-support is proposed. Experimental results show that w-support can be worked out without much overhead, and interesting patterns may be discovered through this new measurement.

The rest of this paper is organized as follows: First, WARM is discussed. Next, we present the evaluation of transactions with HITS [11], followed by the definition of w-support and the corresponding mining algorithm. An interesting real-life example and experimental results on different types of data are given.

This paper is structured as follows section 2 introduces related work. Section 3 introduces ontologies in data mining. Section 4 presents proposed framework and its elements. Section 5

presents experimental results, finally section 6 presents conclusion.

is article guides a stepwise walkthrough by Experts for writing a successful journal or a research paper starting from inception of ideas till their publications. Research papers are highly recognized in scholar fraternity and form a core part of PhD curriculum. Research scholars publish their research work in leading journals to complete their grades. In addition, the published research work also provides a big weight-age to get admissions in reputed varsity. Now, here we enlist the proven steps to publish the research paper in a journal.

II. RELATED WORK

In Data Mining, the usefulness of association rules is strongly limited by the huge amount of delivered rules. To overcome this drawback, several methods were proposed in the literature such as itemset concise representations, redundancy reduction, and postprocessing

The CLOSET algorithm was efficient method for mining closed itemsets. CLOSET uses a novel frequent pattern tree (FP-tree) structure, which is a compressed representation of all the transactions in the database. Moreover, it uses a recursive divide-and-conquer and database projection approach to mine long patterns. Another solution for the reduction of the number of frequent itemsets is mining maximal frequent itemsets [4]. The authors proposed the MAFIA algorithm based on depth-first traversal and several pruning methods as Parent Equivalence Pruning (PEP), FHUT, HUTMFI, or Dynamic Recording. However, the main drawback of the methods extracting maximal frequent itemsets is the loss of information because the subset frequency is not available; thus, generating rules is not possible. Zaki and Hsiao used frequent closed itemsets in the CHARM algorithm [9] in order to generate all frequent closed itemsets. They used an itemset-tid set search tree and pursued with the aim of generating a small nonredundant rule set [6]. To this goal, the authors first found minimal generator for closed itemsets, and then, they generated nonredundant association rules using two closed itemsets. Pasquier et al. [7] proposed the Close algorithm in order to extract association rules. Close algorithm is based on a new mining method: pruning of the closed set lattice (closed itemset lattice) in order to extract frequent closed itemsets. Association rules are generated starting from frequent itemsets generated from frequent closed itemsets. Nevertheless, Zaki and Hsiao [9] proved that their algorithm CHARM outperforms CLOSET, Close, and Mafia algorithms. Most of the existing systems, being generally based on statistical information, most of these methods do not guarantee that the extracted rules are interesting for the user. Both closed and maximal itemsets mining still break down at low support thresholds. Thus, it is crucial to help the decision-maker with an efficient postprocessing step in order to reduce the number of rules & ease of Use.

III. ONTOLOGIES IN DATA MINING

In the early 1990s, ontology was defined by Gruber as a formal, explicit specification of a shared conceptualization [10]. By conceptualization, we understand here an abstract model of

some phenomenon described by its important concepts. The formal notion denotes the idea that machines should be able to interpret ontology. Moreover, explicit refers to the transparent definition of ontology elements. Finally, shared outlines that ontology brings together some knowledge common to a certain group, and not individual knowledge. Ontologies, introduced in data mining for the first time in early 2000, can be used in several ways: Domain and Background Knowledge Ontologies, Ontologies for Data Mining Process, or Metadata Ontologies. Background Knowledge Ontologies organize domain knowledge and play important roles at several levels of the knowledge discovery process. Ontologies for Data Mining Process codify mining process description and choose the most appropriate task according to the given problem; while Metadata Ontologies describe the construction process of items.

IV. ASSOCIATION RULE MINING USING ONTOLOGICAL RELATIONAL WEIGHTS

The concept of association rule proposed support-confidence measurement framework and reduced association rule mining to the discovery of frequent item sets. Much effort has been dedicated to the classical (binary) association rule mining problem since then. Numerous algorithms have been proposed to extract the rules more efficiently. These algorithms strictly follow the classical measurement framework and produce the same results once the minimum support and minimum confidence are given.

In this paper, we introduce w-support, a new measure of item sets in databases with only binary attributes. The basic idea behind w-support is that a frequent item set may not be as important as it appears, because the weights of transactions are different. These weights are completely derived from the internal structure of the database based on the assumption that good transactions consist of good items. This assumption is exploited by extending Kleinberg's HITS model and algorithm to bipartite graphs. Therefore, w support is distinct from weighted support in weighted association rule mining (WARM). where item weights are assigned. Furthermore, a new measurement framework of association rules based on w-support is proposed. Experimental results show that w-support can be worked out without much overhead, and interesting patterns may be discovered through this new measurement. The rest of this paper is organized as follows: First, WARM is discussed. Next, we present the evaluation of transactions with HITS, followed by the definition of w-support and the corresponding mining algorithm. An interesting real-life example and experimental results on different types of data are given.

ORW-ARM generalizes the traditional model to the case where items have weights Ram kumar et al introduced weighted support of association rules based on the costs assigned to both items as well as transactions. An algorithm called WIS was proposed to derive the rules that have a weighted support larger than a given threshold. Cai et al defined weighted support in a similar way except that they only took item weights into account. The definition broke the downward closure property. As a result, the proposed mining algorithm became more complicated and time consuming. Tao et al provided another definition to retain the "weighted downward closure property."

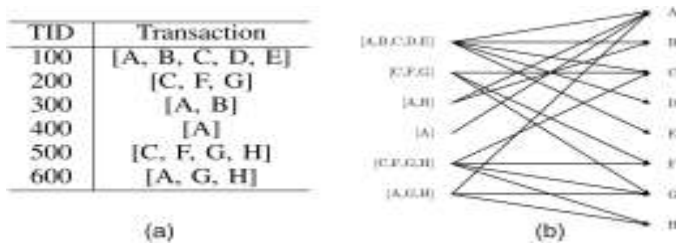


Figure 1: The bipartite graph representation of a database. (a) Database. (b) Bipartite graph.

In conclusion, the methodology of ORW-ARM is to assign weights to items, invent new measures (weighted support) based on these weights, and develops the corresponding mining algorithms.

A directed graph is created where nodes denote items and links represent association rules. A generalized version of HITS is applied to the graph to rank the items, where all nodes and links are allowed to have weights. However, the model has a limitation that it only ranks items but does not provide a measure like weighted support to evaluate an arbitrary item set. Anyway, it may be the first successful attempt to apply link-based models to association rule mining.

A. Algorithm Proposed

In the PROPOSED ORWM (Ontological relational weights measure) algorithm, the first step is to retrieve the set of results to the search query. The computation is performed only on this result set, not across all items.

However, little work has been done on weights by considering the authorities and hubs. Authority and hub values are defined in terms of one another in a mutual recursion. Authority is considered as the number of items with in the transactions. An authority value is computed as the sum of the scaled hub values that point to that transaction. Items relevant to the process of finding the authoritative items known as hubs. A hub value is the sum of the scaled authority values of the items it points to.

The algorithm performs a series of iterations, each consisting of two basic steps:

Authority Update: Update each node's Authority score to be equal to the sum of the Hub Scores of each node that points to it. That is, a node is given a high authority score by being linked to by pages that are recognized as Hubs for information.

Hub Update: Update each node's Hub Score to be equal to the sum of the Authority Scores of each node that it points to. That is, a node is given a high hub score by linking to nodes that are considered to be authorities on the subject.

The Hub score and Authority score for a node is calculated with the following algorithm:

1. Start with each node having a hub score and authority score of 1.
2. Run the Authority Update Rule
3. Run the Hub Update Rule
4. Normalize the values by dividing each Hub score by the sum of the squares of all Hub scores, and dividing each Authority score by the sum of the squares of all Authority scores.
5. Repeat from the second step as necessary.

B. ORWM - Exploration

To begin the ranking, $\forall p$, $\text{auth}(p) = 1$ and $\text{hub}(p) = 1$. We consider two types of updates: Authority Update Rule and Hub Update Rule. In order to calculate the hub/authority scores of each node, repeated iterations of the Authority Update Rule and the Hub Update Rule are applied. A k-step application of the Hub-Authority algorithm entails applying for k times first the Authority Update Rule and then the Hub Update Rule.

i. Authority Update Rule

$\forall p$, we update $\text{auth}(p)$ to be:

$$\sum_{i=1}^n \text{hub}(i)$$

Where n is the total number of items connected to p and i is a item connected to p. That is, the Authority score of a item is the sum of all the Hub scores of items that point to it.

ii. Hub Update Rule

$\forall p$, we update $\text{hub}(p)$ to be:

$$\sum_{i=1}^n \text{auth}(i)$$

Where n is the total number of items p connects to and i is a item which p connects to. Thus a item's Hub score is the sum of the Authority scores of all its relating items.

iii. Normalization

The final hub-authority scores of nodes are determined after infinite repetitions of the algorithm. As directly and iteratively applying the Hub Update Rule and Authority Update Rule leads to diverging values, it is necessary to normalize the matrix after every iteration. Thus the values obtained from this process will eventually converge.

C. Pseudo Code

1. $G :=$ set of items
2. for each item p in G do
3. $p.\text{auth} = 1$ // p.auth is the authority score of the item p
4. $p.\text{hub} = 1$ // p.hub is the hub score of the item p
5. function HubsAndAuthorities(G)
6. for step from 1 to k do // run the algorithm for k steps
7. norm = 0
8. for each item p in G do // update all authority values first
9. for each item q in p.incomingNeighbors do // p.incomingNeighbors is the set of items that link to p
10. $p.\text{auth} += q.\text{hub}$
11. $\text{norm} += \text{square}(p.\text{auth})$ // calculate the sum of the squared auth values to normalize
12. $\text{norm} = \text{sqrt}(\text{norm})$
13. for each item p in G do // update the auth scores
14. $p.\text{auth} = p.\text{auth} / \text{norm}$ // normalize the auth values
15. norm = 0
16. for each item p in G do // then update all hub values
17. for each item r in p.outgoingNeighbors do // p.outgoingNeighbors is the set of items that p links to
18. $p.\text{hub} += r.\text{auth}$
19. $\text{norm} += \text{square}(p.\text{hub})$ // calculate the sum of the squared hub values to normalize

20. $norm = \sqrt{norm}$
21. for each item p in G do // then update all hub values
22. $p.hub = p.hub / norm$ // normalise the hub values.

V. EXPERIMENTAL RESULTS

A. Example 1: consider the bellow transactions as input

TID	TRANSACTIONS
1	1 2 3 4 5
2	3 6 7
3	1 2
4	1
5	3 6 7 8
6	1 7 8

Table 1: Transactional database

From the above transactions bipartite graph representation is

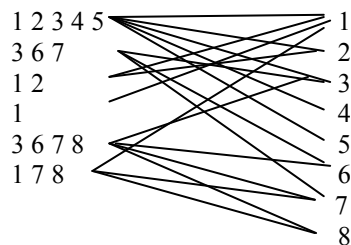


Figure 2: bipartite graph

From the above graph according to our ORWM process compute the authority weights, hub weights and W-support listed in bellow figure 3.

Where vertex weights considered as Authority weights (u) let us perceive that the two update operations described in the figure 3 translated by $v=A^t.u$, $u=A.v$. In order to target the most interesting rules, we gave a support of 0.2 percent; confidence of 7 percent gives number of frequent sets 7 and generates the number of Association rules 2. Those are $\{7\} \rightarrow \{3\}$ with 44.44 confidence, $\{3\} \rightarrow \{7\}$ with 37.03 confidence. For support of 0.2 percent, confidence of 10 percent gives number of frequent sets 3 and generates the number of Association rules 0.

Support (%)	Confidence (%)	No. of Frequent item sets	Number of Association Rules
0-1	9	3	0
0-2	10	3	0
0-2	7	7	2
0-5	8	0	0
0.5	10	0	0

Table 2 Results of ex.1 based on weights

```

The Matrix:
1 1 1 1 1 0 0 0
0 0 1 0 0 1 1 0
1 1 0 0 0 0 0 0
1 0 0 0 0 0 0 0
0 0 1 0 0 1 1 1
1 0 0 0 0 0 1 1
The transpose of matrix:
1 0 1 1 0 1
1 0 1 0 0 0
1 1 0 0 1 0
1 0 0 0 0 0
1 0 0 0 0 0
0 1 0 0 1 0
0 1 0 0 1 1
0 0 0 0 1 1
Vertex Weights:
24
12
18
6
6
12
18
12
Hub Weights:
540
324
216
108
432
324
0.6111111111111112
0.3888888888888889
0.6666666666666666
0.2777777777777778
0.2777777777777778
0.3888888888888889
0.5555555555555556
0.3888888888888889
Press any key to continue . . .

```

Figure 3 display the weights of example 1

B. Example 2

This example proposed to present the efficiency of our new approach concerning the reduction of the number of rules. In this proposed example, we have taken number of items 38 with 768 transactions. We have listed bellow some of the results from the example.

Support (%)	Confidence (%)	No. of frequent itemsets	Number of Association Rules generated
0.1	40	9	13
0.2	12	7	12
0.3	33	7	12
0.5	20	5	4
0.6	65	3	0
0.2	60	7	10
0.1	50	9	12

Table3: example 2 results based on weights

VI. CONCLUSION

This thesis discusses the problem of selecting interesting association rules throughout huge volumes of discovered rules. The major contributions of our paper are stated below. We propose to integrate user knowledge in association rule mining using ontologies. Our ORWM process assigns the weights to the attributes. Mining performed based on the ontological relational weights. By applying our new WARM (weighted Association Rule Mining) approach over a voluminous questionnaire database, we allowed the integration of domain expert knowledge in the postprocessing step in order to reduce the number of rules to several dozens or less.

ACKNOWLEDGMENT

N.RADHIKA would like to thank Asst. Prof. K. Vidya and Prof. Sujatha (Aurora's Technological and Research Institute Hyderabad, India) for their careful reading and valuable suggestion for the improvement of the paper's presentation

REFERENCES

- [1] R. Agrawal, T. Imielinski, and A. Swami, "Mining Association Rules between Sets of Items in Large Databases," Proc. ACM SIGMOD, pp. 207-216, 1993.
- [2] U.M. Fayyad, G. Piatetsky-Shapiro, P. Smyth, and R. Uthurusamy, Advances in Knowledge Discovery and Data Mining. AAAI/MIT Press, 1996.
- [3] M.J. Zaki and M. Ogihara, "Theoretical Foundations of Association Rules," Proc. Workshop Research Issues in Data Mining and Knowledge Discovery (DMKD '98), pp. 1-8, June 1998.
- [4] D. Burdick, M. Calimlim, J. Flannick, J. Gehrke, and T. Yiu, "Mafia: A Maximal Frequent Itemset Algorithm," IEEE Trans. Knowledge and Data Eng., vol. 17, no. 11, pp. 1490-1504, Nov. 2005.
- [5] J. Li, "On Optimal Rule Discovery," IEEE Trans. Knowledge and Data Eng., vol. 18, no. 4, pp. 460-471, Apr. 2006.
- [6] M.J. Zaki, "Generating Non-Redundant Association Rules," Proc. Int'l Conf. Knowledge Discovery and Data Mining, pp. 34-43, 2000.
- [7] N. Pasquier, Y. Bastide, R. Taouil, and L. Lakhal, "Efficient Mining of Association Rules Using Closed Itemset Lattices," Information Systems, vol. 24, pp. 25-46, 1999.
- [8] Toivonen, M. Klemettinen, P. Ronkainen, K. Hatonen, and H. Mannila, "Pruning and Grouping of Discovered Association Rules," Proc. ECML-95 Workshop Statistics, Machine Learning, and Knowledge Discovery in Databases, pp. 47-52, 1995.
- [9] B. Baesens, S. Viaene, and J. Vanthienen, "Post-Processing of Association Rules," Proc. Workshop Post-Processing in Machine Learning and Data Mining: Interpretation, Visualization, Integration, and Related Topics with Sixth ACM SIGKDD, pp. 20-23, 2000. J. Zaki and C.J. Hsiao, "Charm: An Efficient Algorithm for Closed Itemset Mining," Proc. Second SIAM Int'l Conf. Data Mining, pp. 34-43, 2002.
- [10] T.R. Gruber, "A Translation Approach to Portable Ontology Specifications," Knowledge Acquisition, vol. 5, pp. 199-220, 1993.
- [11] Implementing an online version of hyperlink-induced topic search (hits) algorithm Amith Kollam Chandranna, Department of Computer Science San José State University, project thesis.

First Author - N. Radhika, received her B.Tech degree in information technology from Sri Venkateswara Engineering College, JNTU, Hyderabad in the year 2008 and at present she is pursuing (M.Tech), Software Engineering from the year 2009, Department of computer science and engineering, Aurora's technological and research Institute, Hyderabad, India, nandhyala.radhiaka@gmail.com



Second Author - K.Vidya, Sr.Asst.Professor, Department of computer science and engineering, Aurora's technological and research Institute, Hyderabad, India, kammavidya@gmail.com

