Volume 2, Issue 8, August 2012 ISSN: 2277 128X

International Journal of Advanced Research in Computer Science and Software Engineering

Research Paper

Available online at: www.ijarcsse.com

An Implementation of Mining Weighted Association Rules without Preassigned Weights

Arumalla Nagaraju[1] , Yallamati Prakasarao[2], A.Veeraswamy[3]

[1&2]Assistant Professor, Department Of CSE, Chalapathi Institute of Engineering & Technology,
ChalapthiNagar,Lam,Guntur.

[3]Research Scholar, Department of IT, VELTECH Dr.RR& Dr.SR TECHNICAL UNIVERSITY, Chennai, Tamil Nadu.

Abstract*— Association rule mining is a key issue in data mining. However, the classical models ignore the difference between the transactions, and the weighted association rule mining does not work on databases with only binary attributes. In this project, we introduce a new measure w-support, which does not require pre assigned weights. It takes the quality of transactions into consideration using link-based models. A fast mining algorithm is given, and a large amount of experimental results are presented and also compare the support based rules and w-support based rules.

***Keywords*— Association Rule Mining, Data Mining, Pre-processing, Support Based Rules**

I. Introduction

"Data mining refers to extracting or "mining" knowledge from large amounts of data". The term is actually a misnomer. Remember that the mining of gold from rocks or sand is referred to as gold mining rather than rock or sand mining. Thus, data mining should have been more appropriately named "knowledge mining from data," which is unfortunately somewhat long. "Knowledge mining," a shorter term may not reflect the emphasis on mining from large amounts of data. Nevertheless, mining is a vivid term characterizing the process that finds a small set of precious nuggets from a great deal of raw material[1]. Thus, such a misnomer that carries both "data" and "mining" became a popular choice. Many other terms carry a similar or slightly different meaning to data mining, such as knowledge mining from data, knowledge extraction, data/pattern analysis, data archaeology, and data dredging.

Many people treat data mining as a synonym for another popularly used term, Knowledge Discovery from Data, or KDD. Alternatively, others view data mining as simply an essential step in the process of knowledge discovery. "The aim of data mining is to make sense of large amounts of mostly unsupervised data, in some domain."The above statement defining the aims of data mining (DM) is intuitive and easy to understand. The users of DM are often domain experts who not only own the data but also collect the data themselves. We assume that data owners have some understanding of the data and the processes that generated the data. Businesses are the largest group of DM users, since they routinely collect massive amounts of data and have a vested interest in making sense of the data. Their goal is to make their companies more competitive and profitable[2]. Data owners desire not only to better understand their data but also to gain new knowledge about the domain (present in their data) for the purpose of solving problems in novel, possibly better ways.

II. Association Rules Mining

Association rule mining finds interesting association or correlation relationships among large set of data items. With massive amounts of data continuously being collected and stored, many industries are becoming interested in mining association rules from their databases. The discovery of interesting association relationships among huge amounts of business transaction records can help in many business decision making processes, such as catalog design, cross marketing, and loss-leader analysis[3].

The problem of association rule mining is defined as: Let I={i1,i2,…,in} be a set of *n* binary attributes called items. Let D={t1,t2,…,tm} be a set of transactions called the database. Each transaction in *D* has a unique transaction ID and contains a subset of the items in I. A rule is defined as an implication of the form $X \Rightarrow Y$ where $X, Y \subseteq I$ and $X \cap Y = \emptyset$. The sets of items (for short item sets) *X* and *Y* are called antecedent (left-hand-side or LHS) and consequent (right-hand-side or RHS) of the rule respectively.To illustrate the concepts, we use a small example from the supermarket domain. The set of items is I = {milk, bread, butter, beer} and a small database containing the items (1 codes presence and 0 absence of an item in a transaction) is shown in the table to the right. An example rule for the supermarket could be $\{\mathrm{butter}, \mathrm{bread}\} \Rightarrow \{\mathrm{milk}\}$ meaning that if butter and bread is bought, customers also buy milk.

A. *Weighted Association Rule Mining*

The concept of association rule was first introduced in. It proposed the support-confidence measurement framework and reduced association rule mining to the discovery of frequent item sets. The following year a fast mining algorithm, Apriori, was proposed. 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. WARM generalizes the traditional model to the case where items have weights. Ramkumar et al[4].

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[5]. 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." In to items, invent new measures (weighted support) based on these weights, and develop the corresponding mining algorithms. Wang and Su proposed a novel approach on item ranking[2]. 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.

B. *Ranking Transactions with HITS*

A database of transactions can be depicted as a bipartite graph without loss of information. Let D={T1; T2; . . . ; Tm} be a list of transactions and I={i1; i2; . . . ; in} be the corresponding set of items. Then, clearly D is equivalent to the bipartite graph G={D,I,E}. The graph representation of the transaction database is inspiring. It gives us the idea of applying link-based ranking models to the evaluation of transactions. In this bipartite graph, the support of an item i is proportional to its degree, which shows again that the classical support does not consider the difference between transactions[1]. However, it is crucial to have different weights for different transactions in order to reflect their different importance. The evaluation of item sets should be derived from these weights. Here comes the question of how to acquire weights in a database with only binary attributes. Intuitively, a good transaction, which is highly weighted, should contain many good items; at the same time, a good item should be contained by many good transactions. The reinforcing relationship of transactions and items is just like the relationship between hubs and authorities in the HITS model[5]. Regarding the transactions as "pure" hubs and the items as "pure" authorities, we can apply HITS to this bipartite graph. When the HITS model eventually converges, the hub weights of all transactions are obtained. These weights represent the potential of transactions to contain high-value items. A transaction with few items may still be a good hub if all component items are top ranked. Conversely, a transaction with many ordinary items may have a low hub weight[4].

C. *W-support: A NEW MEASUREMENT*

Item set evaluation by support in classical association rule mining is based on counting. In this section, we will introduce a link-based measure called w-support and formulate association rule mining in terms of this new concept[2]. An item set is said to be significant if its w-support is larger than a user specified value. Therefore, w-support can be regarded as a generalization of support, which takes the weights of transactions into account. These weights are not determined by assigning values to items but the global link structure of the database. This is why we call w-support link based. Moreover, we claim that w-support is more reasonable than counting-based measurement.

III. LITERATURE SURVEY

The network structure of a hyperlinked environment can be a rich source of information about the content of the environment, provided we have elective means for understanding it. We develop a set of algorithmic tools for extracting information from the link structures of such environments, and report on experiments that demonstrate their electiveness in a variety of contexts on the World Wide Web. The central issue we address within our framework is the distillation of broad search topics, through the discovery of \authoritative" information sources on such topics[8]. We propose and test an algorithmic formulation of the notion of authority, based on the relationship between a set of relevant authoritative pages and the set of \hub pages" that join them together in the link structure. Our formulation has connections to the eigenvectors of certain matrices associated with the link graph; these connections in turn motivate additional heuristics for link-based analysis.

Computing Hubs and Authorities:

The method of the previous section provides a small sub graph G that is relatively focused on the query topic | it has many relevant pages, and strong authorities. We now turn to the problem of extracting these authorities from the overall collection of pages, purely through an analysis of the link structure of *G*. The simplest approach, arguably, would be to order pages by their in-degree | the number of links that point to them | in G. We rejected this idea earlier, when it was applied to the collection of *all* pages containing the query term; but now we have explicitly constructed a small collection of relevant pages containing most of the authorities we want to find. Thus, these authorities both belong to G and are heavily referenced by pages within *G* [6].

Indeed, the approach of ranking purely by in-degree does typically work much better in the context of *G* than in the earlier settings we considered; in some cases, it can produce uniformly high-quality results. However, the approach still retains some significant problems. For example, on the query "java", the pages with the largest in-degree consisted of www.gamelan.com and java.sun.com, together with pages advertising for Caribbean vacations, and the home page of Amazon Books. This mixture is representative of the type of problem that arises with this simple ranking scheme: While the firest two of these pages should certainly be viewed as good" answers, the others are not relevant to the query topic; they have large in-degree but lack any thematic unity. The basic difficulty this exposes is the inherent tension that exists within the sub graph *G* between strong authorities and pages that are simply universally popular"; we expect the latter type of pages to have large in-degree regardless of the underlying query topic[6].

One could wonder whether circumventing these problems requires making further use of the textual content of pages in the base set, rather than just the link structure of *G* . We now show that this is not the case | it is in fact possible to extract information more effectively from the links | and we begin from the following observation. Authoritative pages relevant to the initial query should not only have large in-degree; since they are all authorities on a common topic, there should also be considerable overlap in the *sets* of pages that point to them. Thus, in addition to highly authoritative pages, we expect to find what could be called hub pages: these are pages that have links to multiple relevant authoritative pages. It is these hub pages that \pull together" authorities on a common topic, and allow us to throw out unrelated pages of large in-degree. (A skeletal example is depicted in Figure 2; in reality, of course, the picture is not nearly this clean.)[7]

Hubs and authorities exhibit what could be called a mutually reinforcing relationship: a good hub is a page that points to many good authorities; a good *authority* is a page that is pointed to by many good hubs.

Clearly, if we wish to identify hubs and authorities within the subgraph *G*, we need a method for breaking this circularity.

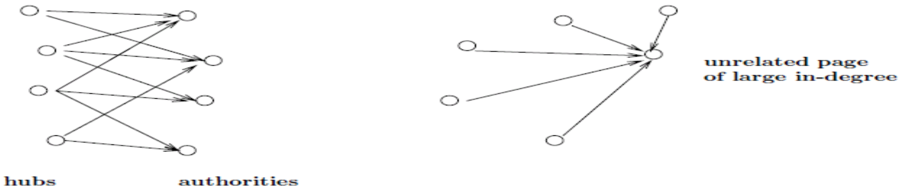


Figure 2.1 a densely linked set of Hubs and authorities

In the HITS 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 Web pages. Authority and hub values are defined in terms of one another in a mutual recursion. An authority value is computed as the sum of the scaled hub values that point to that page. A hub value is the sum of the scaled authority values of the pages it points to. Some implementations also consider the relevance of the linked pages. 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[7].

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:

i) Start with each node having a hub score and authority score of 1.
ii) Run the Authority Update Rule
iii) Run the Hub Update Rule
iv) 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.
v) Repeat from the second step as necessary.

HITS, like Page and Brin's PageRank, is an iterative algorithm based on the linkage of the documents on the web. However it does have some major differences:

It is executed at query time, not at indexing time, with the associated hit on performance that accompanies query-time processing. Thus, the hub and authority scores assigned to a page are query-specific. It is not commonly used by search engines. It computes two scores per document, hub and authority, as opposed to a single score. It is processed on a small subset of 'relevant' documents, not all documents as was the case with Page Rank[7].

.

IV. PROBLEM DEFINITION

Association rule mining aims to explore large transaction databases for association rules, which may reveal the implicit relationships among the data attributes. It has turned into a thriving research topic in data mining and has numerous practical applications, including cross marketing, classification, text mining, Web log analysis, and recommendation systems .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 preassigned weights. However, most data types do not come with such preassigned 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. 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 aim is to get the accurate and better association rules than support based rules. For that we have use HITS and apriori algorithms. The existing system generates rules based on the support count.In proposed system we generate rules by providing hub weight to the transactions by using HITS algorithm, and then apply apriori algorithm, depending on that hub weights we generate association rules.

V. PROPOSED METHODLOGY

A better association rule mining algorithm is proposed where elements will not be assigned weights. A new measurement framework of association rules based on *w*-support is proposed, which does not require preassigned weights. It takes the quality of transactions into consideration using link-based models. 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.

In the proposed system w-support value is calculate by using the hub weights of the transactions. The hub weights are calculated by using the HITS algorithm.

The w-support value is calculate as

$$wsupp(X) = \frac{\sum_{T:X\subset T\wedge T\in D} \; hub(T)}{\sum_{T:T\in D} hub(T)}$$

Where hub(T) is the hub weight of the transaction. The w-support of particular item is division between the sum of hub weights of transactions contain that particular item and sum of hub weights of the all the transactions.

First we have calculated the one item sets and their support count. And then we have calculated the hub weights of the each transaction using HITS algorithm. Then we calculate the support and w-support values for the each item. Next we apply the apriori algorithm on the one item sets and generate two items, three items and so on item sets. After that we give minimum support and minimum support confidence to get the association rules. These are support based results. After that we give minimum w-support and minimum w-confidence to get the association rules. These are w-support based rules. Finally we compare these results.

```
The proposed algorithm is
Start
For all items i initialize auth (i) =0
For (l=0; l<num_it; l++)
{
For each i set auth' (i) =0
For all transaction t belongs to database
{
Hub (t) = sum of all auth (i) where i belongs to t
Auth' (i) +=hub (t) for each item i belongs to t
}
Auth (i) =auth' (i) for each i
Normalize auth}L1= {{i} | wsupp(i)>minsupp}
K=2
While (Lk-1 ≠ )
```

```
{
Ck = Apriori-gen (Lk-1) // Ck contains K-itemsets using Apriori-gen function
For all transactions t D
{
Ct = subset (Ck, t) //check weather Ck belongs to Transaction t
For all candidates c Ct
{
c.wsupp+=hub(t)
}
H+=hub(t)
}
Add Lk to result list
Lk = {c Ck | c.wsupp/h >= minsup}
}
End
```

VI. **EXPERIMENTAL RESULTS**

There are four result sets are implemented in

- Preprocessing.
- Support Based Rules.
- W-Support Based Rules.
- Support vs. W-Support.

A. Pre processing the data set

Preprocessing the data set includes, the transactions data set is loaded. After that we calculate the support count of the each individual item in all the transactions. Next, we calculate the hub weights of the all the transactions by using the HITS algorithm. And then we calculate the support and w-support values for the each and every item. And next we apply the apriori algorithm to get the two item sets, three item sets and so on.

Tid	Trans
1	0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26
2	30,31,32
3	33,34,35
4	36,37,38,39,40,41,42,43,44,45,46
5	38,39,47,48

Table 4.1: Data set format.

The w-support value is calculated by using following formula.

$$wsupp(X) = \frac{\sum_{T:X\subset T \wedge T\in D} hub(T)}{\sum_{T:T\in D} hub(T)}$$

Where hub (T) is the hub weight of the transaction. The w-support of particular item is division between the sum of hub weights of transactions contain that particular item and sum of hub weights of the all the transactions.

B. SUPPORT BASED RULES

After generating the two item sets, three item sets and so on, we calculate the support and w-support values for those item sets. Now we give the minimum support and minimum confidence values. The item sets which have above the minimum support and minimum confidence are the association rules.

C. W-SUPPORT BASED RULES

After generating the two item sets, three item sets and so on, we calculate the support and w-support values for those item sets. Now we give the minimum w-support and minimum w-confidence values. The item sets which have above the minimum support and minimum confidence are the association rules.

The w-confidence value is calculated by using following formula.

$$wconf(X \Rightarrow Y) = \frac{wsupp(X \cup Y)}{wsupp(X)}.$$

The w-support of association rule x=> y is calculated by using following formula.

$$wsupp(X \Rightarrow Y) = wsupp(X \cup Y),$$

D. SUPPORT VS W-SUPPORT

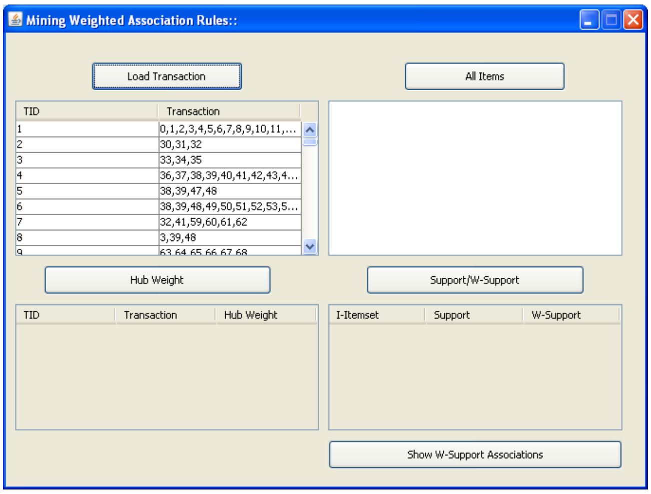
After generating rules for both support and w-support we have compare that results. The difference between the association rules for the sample data set are given below

Support based rules:

```
{41 1327}  ->  {39} (1)  100.0%
{48 170}  ->  {38} (2)  100.0%
{48 36}  ->  {38} (3)  100.0%
{41 36}  ->  {38} (4)  100.0%
{39 110}  ->  {38} (5)  100.0%
{48 110}  ->  {38} (6)  100.0%
{39 48 110}  ->  {38} (7)  100.0%
{39 41 110}  ->  {38} (8)  100.0%
{37}  ->  {38} (9)  100.0%
{170}  ->  {38} (10)  98.14%
{110}  ->  {38} (11)  97.56%
{39 170}  ->  {38} (12)  96.96%
{41 110}  ->  {38} (13)  95.0%
{286}  ->  {38} (14)  93.75%
{36}  ->  {38} (15)  93.18%
{39 36}  ->  {38} (16)  90.0%
{249}  ->  {39} (17)  88.23%
{740}  ->  {39} (18)  88.23%
{39 255}  ->  {48} (19)  86.36%
{48 38 41}  ->  {39} (20)  85.41%
{38 41 110}  ->  {39} (21)  84.21%
{48 41}  ->  {39} (22)  82.17%
{255}  ->  {48} (23)  80.64%
{41 110}  ->  {39} (24)  80.0%
{41 110}  ->  {39 38} (25)  80.0%
{48 89}  ->  {39} (26)  79.16%
{41}  ->  {39} (27)  78.24%
```

W-support Based Rules

```
{41 1327}  ->  {39} (1)  100.0%
{48 170}  ->  {38} (2)  100.0%
{39 48 170}  ->  {38} (3)  100.0%
{41 170}  ->  {38} (4)  100.0%
{48 36}  ->  {38} (5)  100.0%
{41 36}  ->  {38} (6)  100.0%
{39 110}  ->  {38} (7)  100.0%
{48 110}  ->  {38} (8)  100.0%
{39 48 110}  ->  {38} (9)  100.0%
{39 41 110}  ->  {38} (10)  100.0%
{110}  ->  {38} (11)  98.63%
{170}  ->  {38} (12)  97.93%
{39 170}  ->  {38} (13)  96.61%
{41 110}  ->  {38} (14)  94.44%
{41 89}  ->  {39} (15)  92.59%
{36}  ->  {38} (16)  92.4%
{39 36}  ->  {38} (17)  90.56%
```

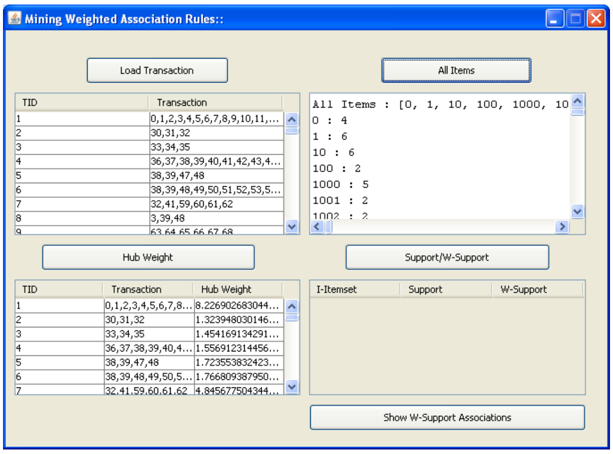

Mining Weighted Association Rules::
Load Transaction
All Items
Hub Weight
Support/W-Support
Show W-Support Associations

TID	Transaction
1	0,1,2,3,4,5,6,7,8,9,10,11,...
2	30,31,32
3	33,34,35
4	36,37,38,39,40,41,42,43,4...
5	38,39,47,48
6	38,39,48,49,50,51,52,53,5...
7	32,41,59,60,61,62
8	3,39,48
9	63,64,65,66,67,68

TID	Transaction	Hub Weight

I-Itemset	Support	W-Support

Fig 1: After loading the data set, we calculate the support value of the each and every single items that are appeared in the data set transactions. Support value means the total number of time that item appeared in the data set transactions


Mining Weighted Association Rules::
Load Transaction
All Items
Hub Weight
Support/W-Support
Show W-Support Associations

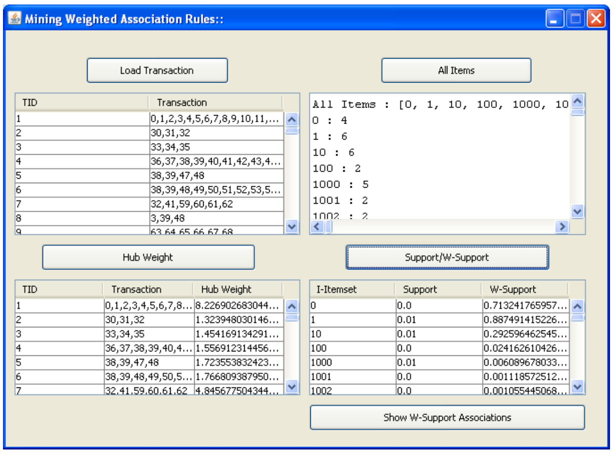
TID	Transaction
1	0,1,2,3,4,5,6,7,8,9,10,11,...
2	30,31,32
3	33,34,35
4	36,37,38,39,40,41,42,43,4...
5	38,39,47,48
6	38,39,48,49,50,51,52,53,5...
7	32,41,59,60,61,62
8	3,39,48
9	63,64,65,66,67,68

```
All Items : [0, 1, 10, 100, 1000, 10
0 : 4
1 : 6
10 : 6
100 : 2
1000 : 5
1001 : 2
1002 : 2
```

TID	Transaction	Hub Weight
1	0,1,2,3,4,5,6,7,8...	8.226902683044...
2	30,31,32	1.323948030146...
3	33,34,35	1.454169134291...
4	36,37,38,39,40,4...	1.556912314456...
5	38,39,47,48	1.723553832423...
6	38,39,48,49,50,5...	1.766809387950...
7	32,41,59,60,61,62	4.845677504344...

I-Itemset	Support	W-Support

Fig 2: Hub weights of the all the transactions


Mining Weighted Association Rules::
Load Transaction
All Items
Hub Weight
Support/W-Support
Show W-Support Associations

TID	Transaction
1	0,1,2,3,4,5,6,7,8,9,10,11,...
2	30,31,32
3	33,34,35
4	36,37,38,39,40,41,42,43,4...
5	38,39,47,48
6	38,39,48,49,50,51,52,53,5...
7	32,41,59,60,61,62
8	3,39,48
9	63,64,65,66,67,68

```
All Items : [0, 1, 10, 100, 1000, 10
0 : 4
1 : 6
10 : 6
100 : 2
1000 : 5
1001 : 2
1002 : 2
```

TID	Transaction	Hub Weight
1	0,1,2,3,4,5,6,7,8...	8.226902683044...
2	30,31,32	1.323948030146...
3	33,34,35	1.454169134291...
4	36,37,38,39,40,4...	1.556912314456...
5	38,39,47,48	1.723553832423...
6	38,39,48,49,50,5...	1.766809387950...
7	32,41,59,60,61,62	4.845677504344...

I-Itemset	Support	W-Support
0	0.0	0.713241765957...
1	0.01	0.887491415226...
10	0.01	0.292596462545...
100	0.0	0.024162610426...
1000	0.01	0.006089678033...
1001	0.0	0.001118572512...
1002	0.0	0.001055445068...

Fig 3: support and w-support values of the 1-itemsets.

VII. PERFORMANCE ANALYSIS

Compared with Apriori the proposed mining algorithm requires an additional iterative procedure to compute the hub weights of all transactions. The database is scanned exactly once in each iteration. Therefore, the convergence rate of the hub weights is critical to the performance. It is clear that HITS converges fast on transaction databases. Generally, three or four iterations are enough to achieve a good estimation, which means that our link-based method works at the cost of three or four additional database scans over the traditional techniques.

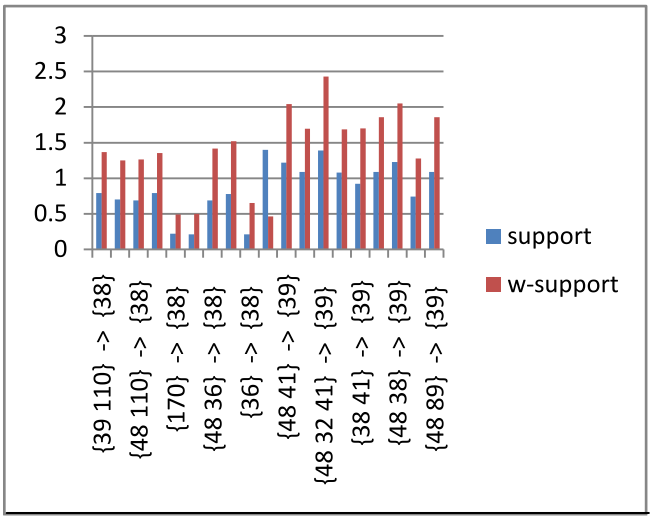


Fig 4: w-supports and supports of the most significant item sets with more than one item. For each item set, the left bar gives its support. Similarly, the right bar represents its w-support.

VIII. CONCLUSION AND FUTURE ENHANCEMENT

This paper has presented a novel framework in association rule mining. First, the HITS model and algorithm are used to derive the weights of transactions from a database with only binary attributes. Based on these weights, a new measure w-support is defined to give the significance of item sets. It differs from the traditional support in taking the quality of transactions into consideration. Then, the w-support and w-confidence of association rules are defined in analogy to the definition of support and confidence. An Apriority-like algorithm is proposed to extract association rules whose w-support and w-confidence are above some given thresholds.

Experimental results show that the computational cost of the link-based model is reasonable. At the expense of three or four additional database scans, we can acquire results different from those obtained by traditional counting-based models. Particularly for sparse data sets, some significant item sets that are not so frequent can be found in the link based model. Through comparison, the paper presented model and method address emphasis on high-quality transactions. The link-based model is useful in adjusting the mining results given by the traditional techniques. Some interesting patterns may be discovered when the hub weights of transactions are taken into account. Moreover, the transaction ranking approach is precious for estimating customer potential when only binary attributes are available, such as in Web log analysis or recommendation systems.

REFERENCES

[1] R. Agrawal, T. Imielinski, and A. Swami, “Mining Association Rules between Sets of Items in Large Datasets,” Proc. ACMSIGMOD ’93, pp. 207-216, 1993.

[2] R. Agrawal and R. Srikant, “Fast Algorithms for Mining Association Rules,” Proc. 20th Int’l Conf. Very Large Data Bases(VLDB ’94), pp. 487-499, 1994.

[3] J.M. Kleinberg, “Authoritative Sources in a Hyperlinked Environment,” J. ACM, vol. 46, no. 5, pp. 604-632, 1999.

[4] O. Kurland and L. Lee, “Respect My Authority! HITS without Hyperlinks, Utilizing Cluster-Based Language Models,” Proc.ACM SIGIR, 2006.

[5] K. Wang and M.-Y. Su, “Item Selection by “Hub-Authority” Profit Ranking,” Proc. ACM SIGKDD, 2002.

[6] G.D. Ramkumar, S. Ranka, and S. Tsur, “Weighted Association Rules: Model and Algorithm,” Proc. ACM SIGKDD, 1998.

[7] Ke Sun and Fengshan Bai, “Mining Weighted Association Rules without Preassigned Weights,” IEEE Transactions On Knowledge And Data Engineering, Vol. 20, No. 4, April 2008.

[8] Authoritative Sources in a Hyperlinked Environment, J.M. Kleinberg, Published by IEEE computer society, J. ACM, vol. 46, no.5, pp. 604-632, 1999.

[9] Mining the Most Interesting Rules, R.J. Bayardo Jr. and R. Agrawal, Published by IEEE computer society, Proc. ACM SIGKDD 99, pp. 145-154, 1999