International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 4 Issue 7 ǁ July. 2016 ǁ PP. 49-55

Most Frequent Item Sets Mining Algorithm Based on MIS-Tree And Multiple Support Array

Chen Xingxing[1], Huang Hongbin[1,2], Du Wenya[1]
[1]*College Of Information Science And Technology, Jinan University ,Guangzhou ,China*
[2]*Zhongshan Aiscent Technologies Ltd, Zhongshan, Guangdong,China*

Abstract: Most Frequent itemsets data mining is the focus and difficulty in the research of the association rules mining. For the united and difficult question to set multiple minimum supports rules mining , this paper proposes a improved algorithm of most frequent item sets mining (MAFP-grow) which is based on combination of multiple item support tree and multiple support array. Firstly it sets parameters MIS dynamically to adjust the threshold of the support, so as to make every item can get a adapted threshold of the support .Then it constructs structure of MIS-tree and support array. At this foundation, when it sets up conditional pattern base, it scans support array without the need for repeatedly scanning MIS-tree .Therefore it reduces the pass of MIS-tree and search space to improves algorithm efficiently. At last , the improved algorithm was used in the movie recommendation system .The experiment result shows that the MAFP-Growth is very effective.

Keywords: *association rules, MIS-tree,multiple minimum supports, conditional pattern base,recommendation system*

I. INTRODUCTION

Association rules[1]are used to show itemsets with high correlation and reveal the dependencies between the things. Frequent itemset mining association rules is one of the core of the Association rules and it is mainly used in association rules, sequence analysis and web log analysis. The Apriori algorithm was introduced in [2] . since 1994,This algorithm mined large itemsets with threshold set by comparing whether itemsets support are greater than threshold. But this may cause a great deal of candidate itemsets and scan the database for more times ,so that it causes great expenses for time and space. To avoid this, Han proposed FP-growth algorithm [3].This algorithm makes use of a divide-and-conquer strategy ,in which a frequent item sets is compressed into frequent pattern tree ,therefore it reduces the times of scan of database. It works in two steps:

1.Make database compressed into frequent pattern tree.
2.Mining all frequent itemsets in the FP-tree.

Compared with Apriori ,FP-grow is more efficient. It doesn’t generate candidate itemsets and scans databases only by two times.

However ,above algorithms mostly improve efficiency by reducing the times of scan of database and space, it can not improve the efficiency of data mining results. Because these algorithm set the single user-specified minimum support(minsup). In most applications, frequency distribution of items are not same or similar , as well as the frequency distribution of items vary widely. Although some frequent distribution of items are low, association rule mining which contains these rule is very important. For example ,in a supermarket, those expensive goods are bought less frequently, but each of them generates more profit. If all data specify single minsup, it can not reflect the frequency distribution of data with different characteristics in the database. It is thus important to capture those rules involving less frequent item.

To solve the above problems, some domestic and foreign scholars have proposed a number of related algorithms. Some of them proposed multi-support model [4] to solve the problem of a single rare item support model emerged , namely on the basis of multi-support model, a lot of data mining algorithms [5] were proposed. They set the minimum support for each item in the database to solve the problem of a single support. But the value of the minimum support per item is pre-set by the user, the size of this value exactly how much it is appropriate is not to set. It is difficult for users with no relevant expertise to set a support for data mining to meet their own needs. If minsup is set too high, those rules that involve rare items will not be found .If minsup is set too low, this may cause combinatorial explosion and take more time for data mining.

Based on multi-item support of the model with the combination of multi-item support tree (MIS-tree) and multi-array support structure, we proposed an improved algorithm for mining association rules , that is MAFP- growth algorithm. This algorithm dynamically adjusts the support threshold so that all items MIS can be a suitable support threshold to solves the problem of each item support threshold set size. Then it construct the multiple support array. When it mines frequent itemsets, it will discovery all frequent itemsets which are in the conditional pattern library by array, and we improve efficiency of algorithm by saving time for MIS-tree traversal time. Based on synthetic data structure[8], the performance of MAFP-growth algorithm is experimented, which is compared with Apriori algorithm, MSapriori [6]and FP-growth algorithm. Algorithm analysis and preliminary tests showed that MAFA-growth algorithm runs more efficiently and has better excavation effects .At last we applied MAFA-growth algorithm in recommended system to test its performance.

II. MIS-tree Structure Model

In order to achieve association rule mining with multiple minimum support , in this paper we adopted the MIS-tree model which is an extension of FP-tree structure and applied to the tree structure [7] of frequent itemsets mining. Firstly, we must know some definition of MIS-tree structure model as follows:

2. 1. MIS-tree Struct

Definition 1:

Let itemset I be $\{I_1,I_2,...,I_n\}$， if any subset of itemset I meets the minimum support, we call it minimum support ,referred to $MIS(I_n)$.

Definition 2:

Let itemset I $=\{I_1,I_2,...,I_n\}$,if $support(I_n) \geq MIS(l_n)$ for any subset , subset I_n is frequent and itemset I is frequent.

Definition 3:

Let MIN is the minimum value of all items MIS as follows: MIN = min [MIS (I_1), MIS (I_2), .., MIS (I_m)].The frequent itemset is collection which its subsets support are greater than MIS.

Example: consider the following items in a database,

I_1,I_2,I_3,I_4.

The items support values are as follows:

$Sup(I_1)=5\%$ $Sup(I_2)=15\%$

$Sup(I_3)=20\%$ $Sup(I_4)=25\%$

The user-specified MIS values are as follows:

$MIS(I_1)=5\%$ $MIS(I_2)=15\%$

$MIS(I_3)=20\%$ $MIS(I_4)=25\%$

$min(MIS(I_1), MIS(I_2), MIS(I_3), MIS(I_4))=10\%$

The following rule doesn’t satisfy its minimum support:

$Sup(I_1)<min$

The following rule satisfies its minimum support:

$Sup(I_2)>min$, $Sup(I_3)>min$, $Sup(I_4)>min$.

So the frequent itemset is $\{I_1,I_2,I_3\}$.

From the above definition, for each of subset, we have assigned to meet its own minimum support constraints. For higher frequency items, we set a higher minimum support. On the contrary, given the lower support constraints, this would resolve the rarity problem of association rule mining.

2. 2. The dynamic adjustment of MIS Value

In our experiment, we need a way to give the MIS value of each item. It takes support of each item as a basis for imparting MIS value by using literature methods[8], the formula is as follows:

$$MIS = \begin{cases} MIS(i) & MIS(i) > MIN \\ MIN & \text{otherwise} \end{cases}$$

$$M(\mathrm{i}) = nf(i)$$

f(i) is the actual frequency of item in the database. MIN is the user-specified minimum item support allowed. The $n(0\leq n\leq 1)$ is parameter that controls how MIS values for items should be related to their frequencies. We need to use two parameters to set the MIS value. When n = 0, we only have a single support, which is the same as traditional association rule mining; when n = 1 and f (i)> MIN, f (i) is MIS value for the item i. The method of adjusting MIS value dynamically is that adjust MIS value of each item in the frequent itemsets , according to formula [Last x (1-MIN),Last x (1+MIN)].Let Last denotes the initial item MIS. MIN is the minimum support. We measure suitable MIS by measuring runtime of algorithm.

2.3 Algorithm MAFP-growth

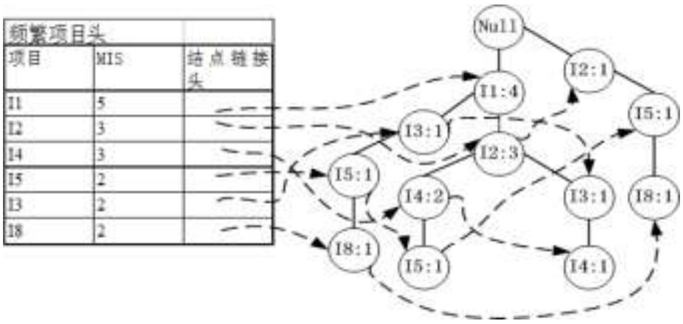
From the process of constructing MIS-tree , we know that all items in the database are compressed into MIS-tree . We mine frequent itemsets by scanning MIS-tree not database . Table 1 shows the transaction database and table 2 shows MIS of each item in the transaction database. The first pass of the algorithm simply counts item occurrences. Each item support is {I_1：4,I_2：4,I_3：2,I_4：3,I_5：3,I_6：1,I_7：1,I_8：2}.Table 2 gives us each item MIS and minimum support MIN=min{MIS(I_1)，MIS(I_2)，MIS(I_3)，MIS(I_4)，MIS(I_5)，MIS(I_6)，MIS(I_7)，MIS(I_8)}=2. Because both support(I_6)and support(I_7) are less than MIN, according to Definition 3, the item{I_6 } and {I_7} are removed from itemsets. In the similar manner, the remaining items which satisfy the Definition are inserted into tree in non-increasing order, as Fig.(1)shows.

Table 1: Transaction Database

Transaction	Item	Item Order
T100	I_1，I_2，I_3，I_4	I_1，I_2，I_4，I_3
T200	I_2，I_5，I_8	I_2，I_5，I_8
T300	I_1，I_2，I_4，I_5	I_1，I_2，I_4，I_5
T400	I_1，I_3，I_5，I_7，I_8	I_1，I_5，I_3，I_7，I_8
T500	I_1，I_2，I_4，I_6	I_1，I_2，I_6，I_4

Table 2: Item MIS

Item	I_1	I_2	I_3	I_4	I_5	I_6	I_7	I_8
MIS	5	3	2	2	2	2	2	2

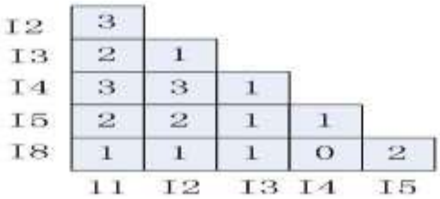


频繁项目头		
项目	MIS	结点链接头
I1	5	
I2	3	
I4	3	
I5	2	
I3	2	
I8	2	

Figure 1 : MIS-tree

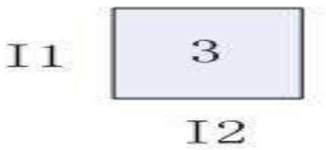
Next, we mine its conditional pattern base[9] for each item. Each item conditional pattern base is relatively independent, and the latter conditional pattern bases obtained do not take advantage of result from former conditional pattern bases, so each of conditional pattern base will need to re-iterate MIS-tree. However, when we seek the whole conditional pattern base for the certain path, this path will be re-iterated, which will cause great waste of time.

Therefore, we improve the algorithm, introducing a two-dimensional array of support. The method of construction of MIS-tree is that the first pass of the MIS-tree simply counts item occurrences, as well as builds two-dimension support array A. If there are n frequent items in the support array A , the number of elements in the array is I = n (n-1) / 2. For example , from Fig.1 we can get the number of frequent items. The frequent items are as follows:I_1:5, I_2:3, I_3:2, I_4:2, I_5:2, I_8:2. When n is 8,the result of the I is 15 , which means that there are 15 elements in the array. The line subscript of array is I_2-I_8 and the column subscript of array is I_1-I_5 , namely A (i, j) = (I_2 ~ I_8, I_1 ~ I_5). In the transaction all frequent items are inserted into MIS-tree at the same time, if the transaction contains {i, j}, the corresponding two-dimensional array A (i, j) is incremented by one. The 2-itemsets[10] support array of the entire data can be drawn as shown in Fig.2.Take I_4 for example, when we want to mine I_4 frequent pattern items, we can get items support from two-dimensional array directly. Therefore ,frequent item I_4 condition pattern library is {I_1, I_2}. According to Definition 2 ,if we want to mine I_4 frequent item , minimum support of itemsets which contains I_4 itemsets must be greater than or equal MIS (I_4). From the Fig.2,we draw that support (I_1I_4) = support (I_2I_4) = 3> MIS (I_4) = 2, the support of $I_1I_4$and I_2I_4 are respectively greater than the minimum support I_4.Therefore $I_1$and I_4 are I_4 frequent itemset . On the basis of this condition pattern library, it will construct A (I_1, I_2) as shows in Fig.3. Consider the association rule $I_1I_2I_4$. Using the itemset support values shown in Fig.3. We draw that I_1I_2 is I4 frequent itemsets. Moreover, MIS-tree and the corresponding support array are recursively generated by MAFA-growth until it has a single path.. The frequent item of the remaining items can be drawn by this method as shown in Table 3.



I2	3				
I3	2	1			
I4	3	3	1		
I5	2	2	1	1	
I8	1	1	1	0	2
	I1	I2	I3	I4	I5

Figure 2: Support array



I1	3
	I2

Figure 3: I4 support array

Table 3 conditional pattern base and frequent itemsets

Item	Conditonal Pattern Base	Frequent Itemsets
I_8	I_1I_8， I_3I_8， I_5I_8， I_2I_8， $I_1I_3I_8$， $I_1I_5I_8$， $I_3I_5I_8$， $I_2I_5I_8$	I_5I_8
I_5	I_1I_5， I_3I_5， $I_1I_3I_5$， I_2I_5， I_4I_5， $I_1I_2I_5$， $I_1I_4I_5$， $I_2I_4I_5$	NULL
I_4	I_1I_4， I2I4， I1I_2I_4， I_3I_4， $I_1I_3I_4$， $I_2I_3I_4$ ， $I_1I_2I_3I_4$	I_1I_4， I_2I_4， $I_1I_2I_4$
I_3	I_1I_3， I_2I_3， $I_1I_2I_3$	I_1I_3
I_2	I_1I_2	I_1I_2

Finally, the largest frequent pattern tree is constructed by frequent itemsets as Fig.4 shows. Every path is a single path in the largest frequent pattern tree, and it's the largest frequent pattern base, so the most frequent itemsets of item I4 is {I_1I_4, $I_1I_2I_4$, I_1I_3, I_5I_8, I_2I_4}.

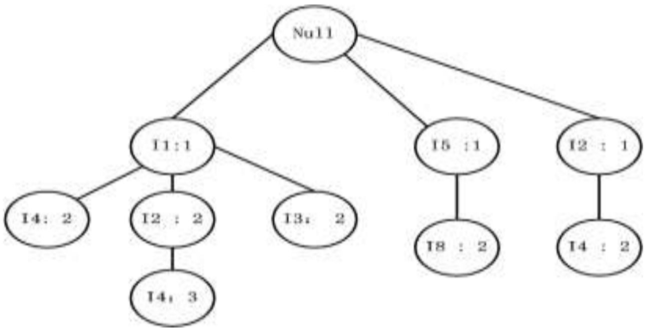


Figure 4: largest frequent pattern tree

Pseudocode：MAFP-growth

```
M=sort(I,MT);
F=init-pass(M,T);
L(i)={<f>|f∈F,f.count ≥MIS（f）}；
Create the root of a MIS-tree ,set root=null;
for(k=2;L(K-1)≠∅;k++)
{
  If          T has a child N such that f. Item=N.item
     Then N.count++;
  else
Create a new node N and set N.count=1;
Let its parent link be linked to it;
}
S=n(n-1)/2;n = number of frequent item;
The array consist of S elements;
Make the first pass over MIS-tree,add N.count to corresponding array;
For each transaction D (i,j)in the array
  If D(i,j)>MIS(i)
  then  j∈frequent item of i;
Call recursion MAFP-growth(T.MIS-tree) until MIS-tree has one path..
```

For sparse data, MAFP-growth algorithm is more efficient. Via MAFP-growth algorithm based on the combination of MIS-tree and two-dimensional support array, we can not only quickly get the support of frequent itemsets ,but also subset of their corresponding support. We do not have to repeatedly perform MIS-tree scan,so as to reduce the overhead of time.

III. Experimental Analysis

The experiment was performed by comparing the runtime of algorithm Apriori, MS-apriori, FP-growth and MAFP-growth . We used the synthetic data T10.I2.D100K as the experimental data ,as well as adopted the different MIS to calculate the runtime. Fig.5 is the result for experiment.

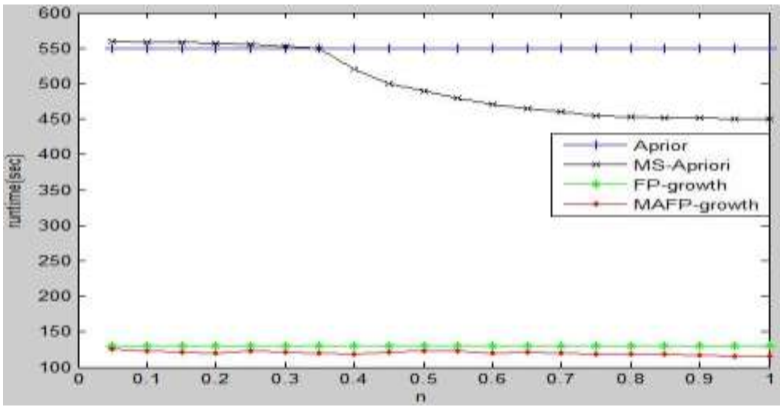


Figure 5 : Runtime of different algorithms

As can be seen from Fig.5, MAFP-growth algorithm is faster ,compared with FP-growth, Apriori and MS-Apriori. Because we can obtain the frequent itemsets via support array without the need to repeatedly scan frequent pattern tree. This can not only reduce the cost of time, but also well reflect the different characteristics of data distribution frequencies .

The experimental data is provided by MovieLens[11]. This data set includes that 943 users score 1683 movies ,which is total of one hundred thousand rating. In this paper ,we mainly adopted parallel method [12]based on MapReduce to complete association rule mining of large-scale data ,as well as used the MAFP-growth in the movie recommendations[13].

Fig. 6 shows the structure diagram of this computing platform. A computer is used as the service node of NameNode and JobTracker, the other three computers are regarded as the service node of DataNode and TaskTracker. The hardware configuration of each node is as follows :CPU model is the Intel Core i5-4790,the memory is 4.00GB, the hard disk is the WD 1TB,and system type is windows 7 .The Hadoop is strictly in accordance with the step and method of Hadoop given by the project website[14] . Moreover ,the hadoop version is 1.1.2.

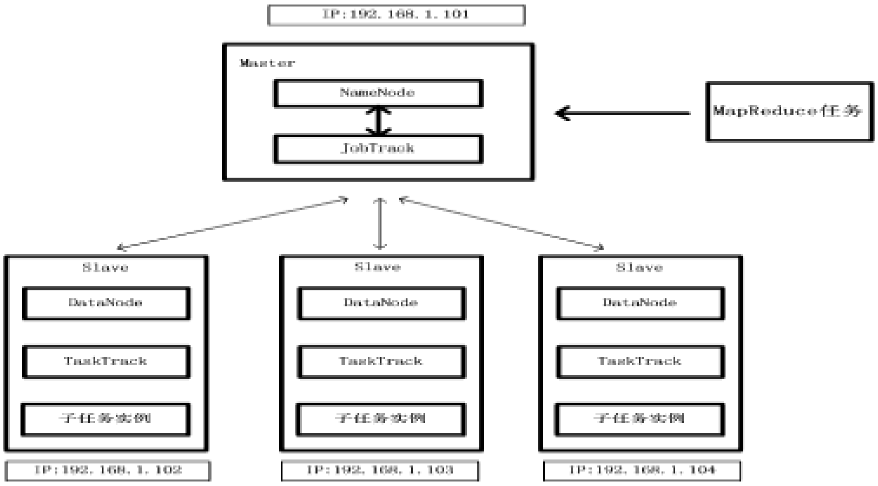
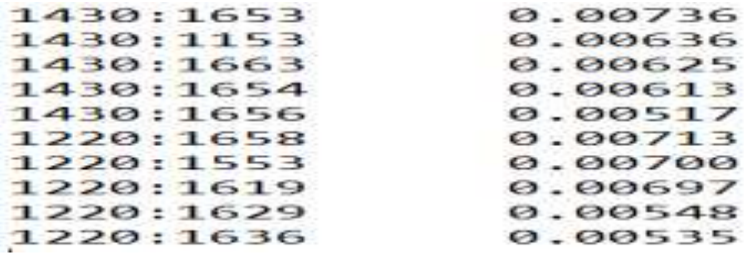


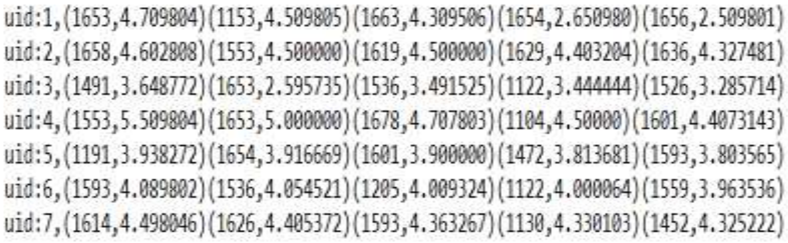
Figure 6: Structure diagram of the hadoop computing platform

To the limited space, Fig.7 and Fig.8 only shows recommendation of part of users with the algorithm MAFP-growth.



```
1430:1653          0.00736
1430:1153          0.00636
1430:1663          0.00625
1430:1654          0.00613
1430:1656          0.00517
1220:1658          0.00713
1220:1553          0.00700
1220:1619          0.00697
1220:1629          0.00548
1220:1636          0.00535
```

Figure 7 : Part of frequent itemsets and the result of item support



```
uid:1,(1653,4.709804)(1153,4.509805)(1663,4.309506)(1654,2.650980)(1656,2.509801)
uid:2,(1658,4.602808)(1553,4.500000)(1619,4.500000)(1629,4.403204)(1636,4.327481)
uid:3,(1491,3.648772)(1653,2.595735)(1536,3.491525)(1122,3.444444)(1526,3.285714)
uid:4,(1553,5.509804)(1653,5.000000)(1678,4.707803)(1104,4.50000)(1601,4.4073143)
uid:5,(1191,3.938272)(1654,3.916669)(1601,3.900000)(1472,3.813681)(1593,3.803565)
uid:6,(1593,4.089802)(1536,4.054521)(1205,4.009324)(1122,4.000064)(1559,3.963536)
uid:7,(1614,4.498046)(1626,4.405372)(1593,4.363267)(1130,4.330103)(1452,4.325222)
```

Figure 8 : The recommendation result based on MAFP-growth

From Fig.7,we observe that when user 1 watches the movie 1430 and scores it highly ,we can consider that user likes this movie. The top five high support associated movie can be filted via the MAFP-growth method . Moreover, the scores of movies for every user are printed as shown in Fig.8. From the experimental results, we can see that we use MAFP-growth in recommendation , which improves the efficiency of algorithm and achieves real-time recommendation .

IV. CONCLUSION

In this paper ,aiming to the traditional association rule without solving the problems associated with rare items, we proposaled MAFP-growth algorithm with a combination of MIS-tree structure and two-dimensional array of support. And we apply the MAFP-growth on the computing platform of Hadoop cloud with the MapReduce programming model to implement parallelized computing and recommendation. From the experimental results shown, Algorithm MAFP-growth has greatly improved in accuracy and mining speed. However, since the data format in the experiment is relative simple ,and there is few influence condition, the more complex model needs to be established in the face of more subsection and more precise operation.

ACKNOWLEDGEMENT

We are thankful to Strategic Emerging Industries Project of Guangdong Province for providing financial support through Award No.2011912027

REFERENCES

[1]. Zhou Tao: Summary of Association Rules Algorithm[J].Intelligence,2011(16)

[2]. Hong-ying, CAI Le-cai and LI Xian-jie: Apriori Association Rules Mining Algorithm Summary[J]. Journal of Sichuan University of Science & Engineering.2011(1).

[3]. Han J.W,Kamber M.Data Mining:Concepts and Techniques.San Francisco,CA:Morgan Kaufinann;2001

[4]. Han,J and Fu,Y .”discovery of multiple-level association rules from large database”VLDB-95

[5]. YAN Jie;QI Wen-Juan;GUO Lei;HUANG Shu-Cheng.Based on Multiple Minimum Supports of Association Rules in Data Mining[J]. Computer Systems & Applications.2014(3).

[6]. Bing Liu,Wynne Hsu and Yiming Ma: Mining Association Rules with Multiple Minimum Supports[J]ACM SIGKDD International Conference on Knowledge Discovery& Data Mining (KDD-99), August 15-18, 1999, San Diego, CA, USA.

[7]. Chang HAO:Association Rule Mining with Multiple Minimum Supports, Northwest University ,2007-05-01

[8]. Gosta Grahne, Menber,IEEE,and Jianfei Zhu, Student Menber,IEEE.Fast Algorithms for Frequent Itemset Mining Using FP-trees[J]IEEE TRANSACTIONS ON KNOWLEDGE AND DATA Engineering.2005,10(17);13347-1362

[9]. ZHOU Qin-liang,LI Yu-chen,GONG Ai-guo: New algorithms for effectively creating conditional pattern bases of FP-Tree[J], Journal of Computer Applications, 2006,6（26）;1418-1421

[10]. WangFan.Research On Maximum Frequent Itemsets Based On Fp-Tree[M].Guangxi University.2013-05-24.

[11]. MovieLens Dataset. http://www.cs.umn.edu/Research/GroupLens/index.html . 2011

[12]. J Polo,DCarrera,et al.Performance-driventaskco-schedulingfor mapreduce environments. Proc of IEEE/IFIP NetworkOperations and Management Symposium . 2010

[13]. FangLuLu, Design and Implementation of Recommendation System Based on Hadoop[D], Beijing University of Posts and Telecommunications ,China, 2015

[14]. Konstantin Shvachko,Hairong Kuang,Sanjay Radia,Robert Chansler.The Hadoop Distributed File System.2010