Research On The Problem Model Of GUI Based On Knowledge Discovery In Database

Wanjiang Han, Tianbo Lu, Yi Sun
School Of Software Engineering, Beijing University of Posts and Telecommunication
Beijing, China
e-mail: hanwanjiang@bupt.edu.cn

Ye Li, Xiao Han, Weijian Li, Chi Liu
International School, Beijing University of Post and Telecommunication
Beijing, China
e-mail: 2011212919@bupt.edu.cn

***Abstract*—In order to improve product quality, it is necessary to summarize, analyze and resolve problems. This paper gives a concept of Problem model, which is essentially the use of Knowledge Discovery in Database (KDD) process. The Problem model derived from problem analysis and summarizing can find problems and the association rules between different problems. The model can improve the development process, improve testing design and find more problems. According to the analysis on Problem data of GUI, this paper established the Problem Model of GUI. Also the model is applied in some projects and got a nice result.**

Keywords-Problem Model; Knowledge Discovery in Database (KDD); Data analysis; Association rules; Apriori algorithm

I. INTRODUCTION

With the rapid development of networks and information technology, product quality is becoming more and more important. There have been lots of technologies and methods to improve quality. This paper proposed the concept of Problem model and analyzed Problem data generated from test results of GUI of communication system. Problem model can be used to improve product quality effectively. It is a new idea of improving quality. And it is also a new research.

3200 test data are collected from our laboratory. The data is subjected to the Knowledge Discovery in Database to extract knowledge.

Furthermore, this knowledge can be used for better prediction and finding Problem. This knowledge can also give some advices and suggestions in design.

II. KNOWLEDGE DISCOVERY IN DATABASE

Knowledge Discovery in Databases (KDD) is a growing research field which consists of many knowledge discovery methods, including quantitative approaches, visualization techniques, and classification approaches.

KDD is useful for analyzing large quantities of data. It has widespread applications.

The overall KDD process consists of several steps and phases is illustrated in Fig. 1 [1, 2] .

The data is subjected to the KDD processes such as Data Cleaning, Data Integration, Data Selection and Data Transformation, for effective knowledge generation.

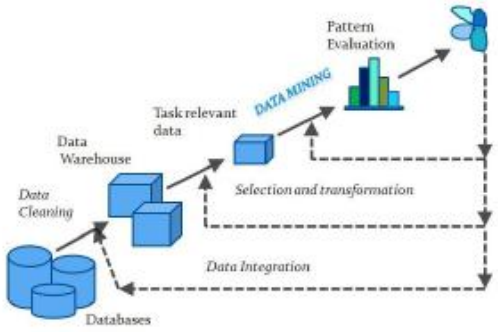

Figure 1. Knowledge Discovery in Databases (KDD)

In Data Cleaning, the irrelevant data are removed from the collected data. In Data Integration, multiple sources may be combined into a common source. The Data Selection process is involved with the selection of data relevant to the analysis and extracting them from the integrated data. The selected data is transformed to the appropriate form for the mining procedure. The process of extracting useful and implicit information from the transformed data is referred to as Data Mining.

Data Mining is the Knowledge Discovery phase and it is the process of extracting implicit, useful, previously unknown, non-trivial information from data. The techniques involved in Data Mining are grouped as Classification, Clustering, Association Rules and Sequences. Association Rule Mining uncovers relationships among data in a database. The quality of an Association Rule is measured by using its support and confidence values and several efficient methods are developed.

A. Association Rule Mining

Association Rule mining was Proposed by Agrawal et al in 1993 [5]. It is an important data mining model studied extensively by the database and data mining community. Assume all data are categorical. Its task is to find certain relationships among a set of data (item set) in the database.

An association rule is a pattern that states when X occurs, Y occurs with certain probability [6].

Association rules mining is interested in finding frequent rules that define relations between unrelated frequent items in databases, and it has two main measurements: support and confidence values. Confidence values are measurements of

rule's strength, while support value corresponds to statistical significance.

Frequent items are defined as the itemset that have support value greater than or equal to a minimum threshold support value, and frequent rules as the rules that have confidence value greater than or equal to minimum threshold confidence values. These threshold values are assumed traditionally to be available for mining frequent itemsets. Association rules Mining is all about finding all rules whose support and confidence exceed the threshold, minimum support and minimum confidence values.

Association rules mining proceeds on two main steps. The first step is to find all Frequent itemsets with adequate supports and the second steps is to generate Association rules by combining these frequent itemsets.

B. *Apriori Algorithms*

Apriori algorithm [7] has been shown as a classical association rule in mining algorithms that have been cited so far, which is used to find frequent item sets in a database and to generate Association Rules from the frequent item sets.

The core issue of the Apriori Algorithm is how to regenerate all the frequent itemsets. It uses an iterative approach layer by layer, applying K-Itemset to search for (K+1)-Itemset. Firstly, the algorithm identifies the collection of Frequent1-Itemset, denoted by L1.Collection of Frequent 2-Itemset "L2" is computed from L1. We use the same method to acquire L_{k+1} (K>1) until it cannot find L_k. Doing this repeatedly, so all the frequency are set out. These rules generated must meet the minimum support and minimum confidence.

Apriori Algorithm adopts recursive method to generate the required frequent sets. Input values are transaction database D and minimum support threshold value min_sup.

III. THE PROCESS OF DEVELOPING PROBLEM MODEL

This paper proposed the concept of Problem model. The model concentrates on Network Management system which meets different demands of networking industry users with its open compatibility and high reliability of the design. The system consists of some components, such as Configuration management, Fault management, Safety management, Maintain management, Performance and so on.

Problem model is based on analyzing and summarizing test data. This model can effectively improve the process of product integration and product test. As a result, the quality of product is improved [3].

KDD is used to generate meaningful results from data and hence it is applied on test data of GUI of communication system to generate knowledge. The knowledge is called Problem Model for GUI of communication system.

The test Data was taken as a raw data and the preprocessing phase of the KDD process was applied on the data to generate transformed data that could be analyzed to extract knowledge from the data. The process consists of few steps including Data Cleaning, Data Integration, Data Selection, Data Transformation, Data Mining, Generation of Patterns, and Knowledge Interpretation.

The following describes processes of Developing Problem Model.

A. *Problem Data Cleaning*

In order to obtain accurate experimental results, we have to process the sample data in advance for subsequent steps. The data are manually collected by testing, so erroneous data and missing values are inevitably generated from data entry errors. We employ data preprocessing methods to smooth out errors, to fill in the missing values, to eliminate redundant data, and to remove inconsistent data.

B. *Problem Data Selection*

3200 test data are divided into some kinds of parts by Problem type. Fig 2 shows the distribution of problem data.

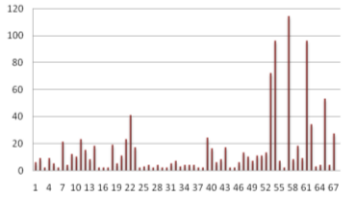

Figure 2. The distribution of problem data.

From Fig. 2, we can see the percent of GUI problem is 65, So we focus on GUI test data. And our model is developed from the problem data of GUI.

C. *Problem Data Transformation*

This step is a quantitative process. We consider problem item of GUI as itemset, and define ID for every item, as table I shows.

TABLE I. ID OF PROBLEM ITEM FOR GUI

Type	Item	ID
accuracy	UI display problems	1
	UI prompt problems	2
	Incorrect spell problems	3
	Punctuation format problems	4
	Translation problems	5
	Chinese in the English version	6
	English in the Chinese version	7
	Interface block problems	8
Ease of use	Learnability problems	9
	Sequencing problems	10
input	Boundary-value problem	11

	Equivalence class problem	12
Design	Design rationality problem	13
	Consistency problem	14
	The date selection problem	15
Other	The progress bar problem	16

D. Problem Data Mining

This step is mining process. Given a transaction data set T, a minimum support and a minimum confident, the set of association rules existing in T is uniquely determined.

Now we consider problem data of product as transaction data set, and define TID for every transaction data set. There are 18 product versions, and Each version has a unique TID, such as 1,2,3,……,18. For example, {1,2,13,14} is a transaction data for TID 1 which means {UI display problem,UI prompt problem,Design rationality problem, Consistency problem}. Similarly, we got Transaction Database T, shown as table II.

TABLE II. TRANSACTION DATA SET T

TID	Itemset
1	1 2 13 14
2	2
3	1 2 9 14
4	1
5	1 2 6 10
6	1 8
7	1 2 6 7 11 14
8	1 2 11 14 16
9	2 3 16
10	1 6 9 13
11	1 9 11 12 13
12	1 2 11
13	1 2 8 9 11 14
14	2 11 13
15	6 11
16	1 2 3 6 8 9 10 11 13 14
17	8 9
18	2 3 8 9 11 12 13 14 15

Given the threshold of minimum support and the threshold of minimum confident as follow:

- min_sup=40%
- min_conf=60%

We use Apriori algorithm to generate frequent itemsets as shown in table III. Fig.3 is also the frequent itemsets. And Generate association rules from frequent item set, as table IV.

TABLE III. FREQUENT ITEMSETS

No	Frequent item sets	Concrete problem	number of times
1	1 2	UI display problems，UI prompt problems	14
2	1 14	UI display problems Consistency problem	13
3	1 9	UI display problems Consistency problem	9
4	1 6	UI display problems Problem of Chinese in the English version	9
5	1 11	UI display problems，Boundary-value problem	11
6	2 14	UI prompt problems，Consistency problem	15
7	2 8	UI prompt problems，Interface block problems	9
8	2 11	UI prompt problems，Boundary-value problem	13
9	14 9	Consistency problem，Learnability problems	9
10	14 8	Consistency problem Interface block problems	10
11	14 11	Consistency problem，Boundary-value problem	13
12	9 11	Learnability problems Boundary-value problem	9
13	1 2 14	UI display problems，UI prompt problems，Consistency problem	12
14	1 2 11	UI display problems，UI prompt problems，Boundary-value problem	9
15	1 14 11	UI display problems，Consistency problemBoundary-value problem	9
16	2 14 8	UI prompt problems Consistency problem Interface block problems	9
17	2 14 11	UI prompt problems Consistency problem Boundary-value problem	11

We focus on strong Association rules. Now we Analyze the frequent itemsets and Association rules：

A. Frequent Itemsets Analysis

From Fig.3, we know that there are 17 frequent itemsets.

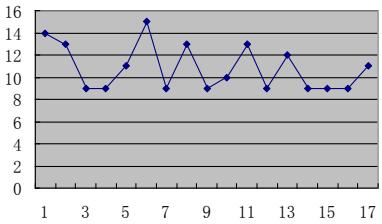

Figure 3. Frequent itemsets

According to the property of Apriori algorithm , the association rules model of processes sequence has the following properties [4]:

Property 1:All the non-empty sub-sequences of frequent operation sequences are frequent.

Property 2:The superset of non-frequent process sequence is non-frequent for sure.

Applying these theory, we know that {UI prompt problem, Consistency problem},{UI prompt problem,Interface block problem},{Consistency problem,Interface block problem} are also frequent itemsets, due to {UI display problem,Consistency problem,Interface block problem} is frequent itemsets.

B. Association Rules Analysis

As Fig 4.shows, we know there are 37 Association rules, Our Goal is to find all the rules that satisfy the user-specified minimum support(minsup) and minimum confidence (minconf).

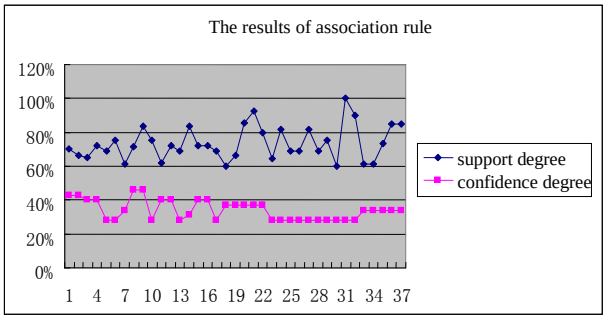

Figure 4. Association rules

In Fig.4, the lower line is support degree, and the upper line is confidence degree.

As Fig.4 shows, there are 22 Association rules that meet our goal.

For the rule of "UI display problems $\Rightarrow$ UI prompt problems", We know that, its confidence degree is 70%,and its support degree is 43%. It tells us that it is 70 percent that UI display problems and UI prompt problems comes out in the same time. And It is 43 percent that UI prompt problems come up while UI display problems comes. So we should pay more attention to them.

For rule 31, the confidence degree is 100%,but its support degree is smaller than min_sup. We also should pay more attention to these Association rules. It also tells us some important message.

C. Problem Model for GUI

In summary, we get the Problem Model for GUI in table IV. They are the strong Association rules that meet our demands. The model tells us the problems and the association problems about GUI of Network Management system. For example, if there is a problem of UI display, we should pay more attentions on problems of UI prompt, coincidence and Boundary.

TABLE IV. PROBLEM MODEL FOR GUI

Type	Items of problem	Association problems
Correctness	UI display problem	UI prompt problem 、coincidence problem、Boundary problem
	UI prompt problem	UI display problem、Consistency problem、Boundary-value problem、Interface block problem
	Translation problems	Incorrect spell problem
	Chinese in the English version	UI display problem
	English in the Chinese version	UI display problem
	Interface block problem	UI prompt problem、Consistency problem
Ease of use	Learnability problem	UI display problems、Consistency problem、Boundary-value problem
Input	Boundary-value problem	UI display problemsUI prompt problems、Consistency problem
	Consistency problem	UI display problem,UI prompt problem,Boundary-value problem,Interface block problem

IV. THE APPLICATION OF PROBLEM MODEL

This model can give us some help in developing and testing.

A. Some Advices In Test Process

The problem model can be used on designing testcase. We should develop more test cases in the items of problems. And we should add more test steps of association problems to every testcase of the items of problems. So the problem model can cover more parts of requirement. This can improve the quality of testing and raising the efficiency of testing. For examples:

- IF there is a problem of UI display during testing, we should pay more attention on the problem of UI prompt, Consistency, Boundary-value.
- IF there is a problem of UI prompt during testing, we should pay more attention on the problem of "UI display problems" , "Consistency problem", "Boundary-value problem" ,"Interface block problems".

The Problem model has applied some projects. In these projects, the testcase has been improved, and we have found more problems. Fig 4 shows the difference between projects applying model and projects no-applying model. So more problems were found.

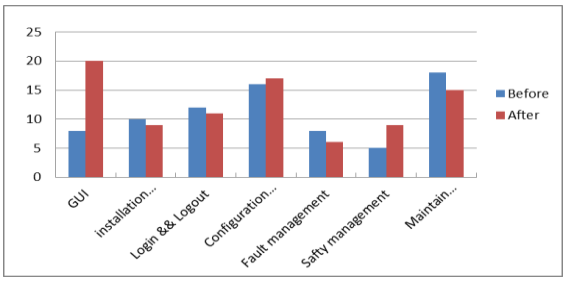

Figure 5. Comparison between problems found before and after the model

B. *Some Advices In Developing Process*

Through Problem Model, we see that GUI problems are centralized in the consistency. Coupling ratio should be decreased by applying strong association rules. For instance, the reduction of coupling showing in Problem Model lessens the present of any relevant problem. And the experiment has shown this as well. Now we take UI-display- problem and UI-promp-problem as an example, Table V shows a comparison between the problem relativity before and after the improvement which makes the primary problem relativity reduce enormously.

TABLE V. COMPARISON BETWEEN BEFORE AND AFTER IMPROVEMENT

Problems	**Related Problems before**	**Related Problems after**
UI display problem	UI prompt problem,Consistency problem,Boundary-value problem	UI prompt problem
UI prompt problem	UI display problem, Consistency problem,Boundary-value problem,Interface block problem	UI display problem,Interface block problem

V. CONCLUSION

A brief study of test data of GUI of Network management system is presented in the paper. The test data was analyzed. The KDD steps were explained and were applied on test data to convert the raw data into a transformed data that can be used for generating more knowledge.

As a result of KDD applied to test data, the Problem Model of GUI was provided by this paper.

The model is useful in developing product and testing product. It can optimize product design and test case. In the end, the product quality can be improved.

With more data to be collected, the model can be improved constantly. We will further extend this model with other areas.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61170273).

REFERENCES

[1] Glenford J. Myers,Tom Badgett,Todd M. Thomas,Corey Sandler, The Art of Software Testing, 2004.

[2] Bender, Requirements Based Testing Process Overview,2009.

[3] Long H,Lee H Y.A New Visualization Technique for Knowledge Discovery in OLAP. Proceedings of the First Pacific-Asia Conference on Knowledg e Discovery and Data Mining,1997.

[4] Jaiwei Han,Michelinne Kamber, Data Mining:Concepts and Techniques,Morgan Kaufmann Publishers, Second Edition,2006.

[5] Margaret H.Dunham,Data Mining:Introductory and Advanced Topics,Pearson Education,2007.

[6] Dion H.Goh and Rebecca P.Ang. An Introduction to Association rule mining: An application in counseling and help-seeking behavior of adolescents. Behaviour Research Methods,39(2),2007,pp.259-266.

[7] Li C, Fan M(2004) Generating association rules based on threaded frequent pattern tree. Comput Eng Appl 4:188-192(iin Chinese).