Pattern Mining and Querying in Business Intelligence

Nittaya Kerdprasop, Fonthip Koongaew, Phaichayon Kongchai,
Kittisak Kerdprasop

Data Engineering Research Unit, School of Computer Engineering,
Suranaree University of Technology, 111 University Avenue,
Nakhon Ratchasima 30000, Thailand
nittaya@sut.ac.th

Abstract. Business intelligence is currently practiced in many organizations through the data mining tools. The most applicable data mining tasks are classification and association rule discovery. Even though existing data mining tools can successfully discover rules that represent relationships among operational stored data, we observe that the number of discovered rules is somewhat exceeding users' need. We therefore propose the extensions of conventional classification and association mining by incorporating search constraints over the discovered rules. The framework of the proposed business intelligence system, as well as its implementation and applications are presented in this paper.

Keywords: Patter mining, Pattern querying, Business intelligence, Logic and constraint programming.

1 Introduction

The term *business intelligence* (BI) has been introduced since 1958 [9] as an actionable goal obtained from insight understanding of interrelationships that exist among stored facts. Currently, BI is a broad term normally refer to any aspect of computer-based business applications including decision support, information management, marketing automation, and intelligent data analysis [5], [6], [7], [11].

BI software in the market contains more or less some modules of intelligent analysis such as statistical-based learning, data mining, to extract useful information hidden in the enterprise databases. The extracted information from this automatic process is however tremendous in its quantity. Analysts have to thoroughly explore and select only the most informative knowledge reported to executives.

We thus propose to add pattern mining and querying functionality to the current BI software to help analysts filter from large amount of knowledge the most relevant ones to their interest. The design and implementation of our pattern mining feature are based on the decision-tree classification and association rule induction. Querying the induced rules can be performed through the logic-based language based on the Prolog syntax. The mining steps of our system are constrained by preferences, which are identified by analyzer. The experimental results confirm the benefit of our system.

2 Pattern Analysis Framework and its Implementation

We design the inductive pattern analysis framework (Figure 1) with the main purpose of providing suggestion to strategic planners helping them making an actionable plan. The content management module is the part to access customer details from the stored data. Not every single detail is to be used by the inductive module, therefore the three modules (i.e., content segmentation, content conversion, and DM format manager) are necessary for screening and extracting potentially useful features from the database. Selected data are then sent to the knowledge management module to analyze and induce model that can characterize customers‟ patterns and predict future events from current situation. These induced models are considered valuable knowledge that will be finally sent back to generate actionable suggestion to the strategic planners.

We implement knowledge mining engine, which is the main part of the proposed business intelligence framework, with the constraint-based logic programming paradigm using Eclipse 6.0 constraint system. The implementation of decision-tree induction [12] with constrained search facility is shown in Figure 2, whereas the constrained association rule mining [1] is illustrated in Figure 3.

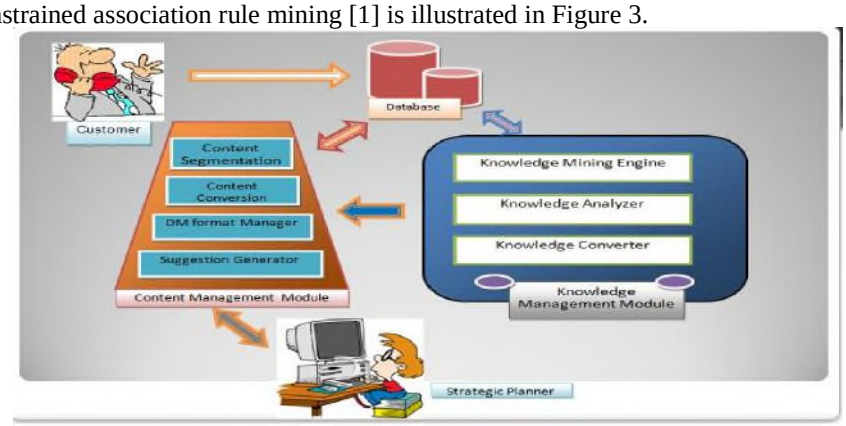

Fig. 1. A framework of knowledge-mining-based intelligent system.

```
% Decision Tree : id3
:-lib(listut). %can use only 1 library
:- lib(sd).
:-dynamic rule_me/1.
:-dynamic allrule/1.
:-dynamic r/1.
% Compile, then run with command :  run.
%  findRule([]).  % to show all rules
%  findRule([q31-yes]).  % show rules containing q31-yes
run :- retractall(rule_me(_)),retractall(allrule(_)),retractall(r(_)),
compile("//D/1-Journal-conf-Publications/Conferences/2012-7-JUL/Data/Social-Sc-Data/id3-Fonthip
ied/data-prolog.txt"),data(Data+Attrs),
main(Data,Attrs,[]),retractall(rule_me([_])),findall(X,rule_me(X),R),assert(allrule(R)).

main([],_,_).
main(_,[],_).
main(Data, Attrs,OldAttr):-
    all_info(Data+Attrs, R1),
    (\+hasOneClass(R1)->
    (maplist(avg_info,R1,Out),
    chooseMin(Out, nil/11.00), OO=O/_,
    [listut]:delete(Attrs,O,NewAttrs),
    (foreach(X,O),param(Data,NewAttrs,OldAttr) do
        (filterData([X],Data,NewData), append_me(OldAttr,[X],OAL), main(NewData,NewAttrs,OAL))
    )); getlast_goal(Data,Att-Ans),
    append_me(OldAttr,[Att-Ans],NOAL),split_rule(NOAL)),assert(rule_me(NOAL)).

findRuleOr([H|T]):- findRule2(H),    findRuleOr(T).
findRuleOr([]) :-    setof(H,r(H),S),split(S).
split([]).
split([H|T]) :- split_rule(H), split(T).
findRule2(X):-  allrule(L), findRule2(X,L).
findRule2(_,[]).
```

Fig. 2. An implementation of decision-tree induction with constrained search.

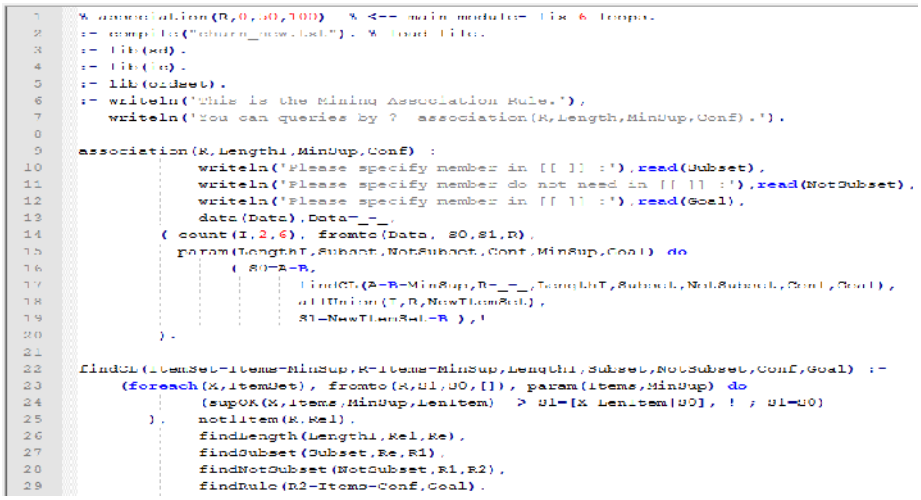

Fig. 3. An implementation of constrained association rule mining.

4 Application to Customer Churn Prediction

Customer relationship management is the process of differentiating valuable customers from the normal ones with the main objective of retaining high quality customers [13]. From the customer churn analysis results, marketing personnel have to assess appropriate efforts for retention campaign. Business sectors take customer churn as a serious subject because the cost of retaining current customers is much lower than acquiring the new ones [13]. Conventional statistical methods such as logistic regression analysis [10] are normally adopted to analyze and predict churning customers. In the context of business intelligence, other emerging techniques from the data mining research field can also be applied to help analyzing churn customer. We propose to extend existing association mining method to facilitate general users for querying mining results in various aspects.

On running the implemented program, we use the churn data in telecommunication industry [3], [8]. The data set contains information of 3333 customers. In the original data set, each customer record has 21 features (or variables) in which the last one is the label churn/non-churn. The first step is feature selection. We then performed a series of eight experiments on the selected churn data set to induce association rules with various constraints:

Exp. 1: Rules are to be induced with thresholds: minimum support = 50 (that means there must be at least 50 records from the total of 3333 satisfying the rule's content) and minimum confidence = 80%. (The other experiments also specify the same minimum support and confidence.)

Exp. 2: Rules must contain the feature churn_False (that is, customer is non-churner).

Exp. 3: Rules must have at least three features.

Exp. 4: Rules must NOT contain the feature "churn_False'.

Exp. 5: Rules must contain the feature "churn_False' at the then-part of the rule.

Exp. 6: Rules must contain either the feature "churn_False", or "churn_True".
Exp. 7: Rules must contain both the feature "churn_True" and "vMailPlan_no".
Exp. 8: Rules must have at least three features, must contain both "churn_False" and "vMailPlan_no", must NOT contain either the feature "vMailMessage_0", or "intlCalls_2", and the target clause of the rules must be "churn_False". Running result of this experiment is shown in Figure 4.

We comparatively illustrate the experimental results in Figure 5.

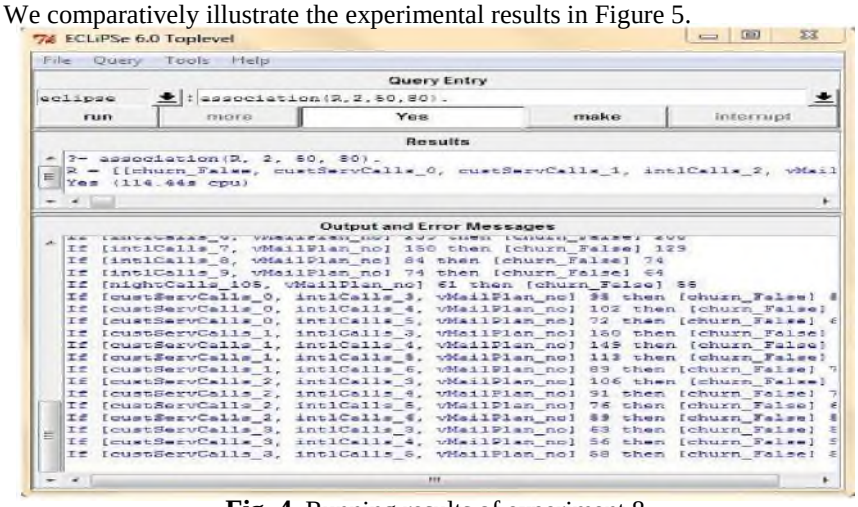

Fig. 4. Running results of experiment 8.

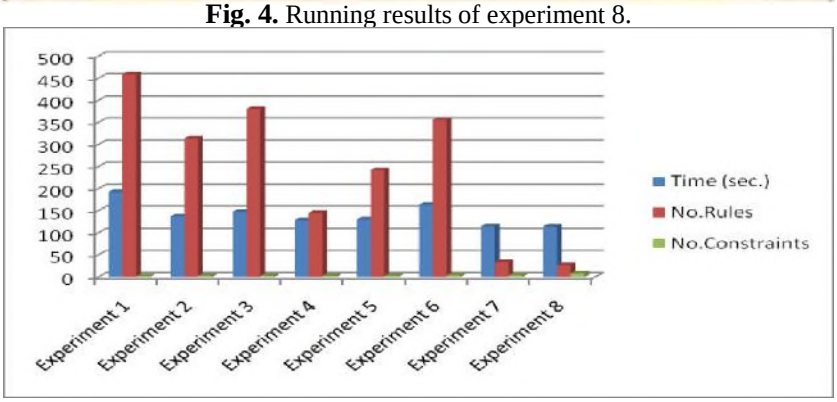

Fig. 5. A comparison of computational time usage and number of rules received from varying constraints in each of the eight experiments.

5 Application to Survey Analysis

The task of automatically extracting patterns from data related to decision making is normally done by applying statistical techniques [4]. Such methods cannot keep pace with the exponential growth of electronic data. Business analysts are gradually exploiting a faster tool of data mining techniques. Among various available data

mining techniques, decision tree induction is the most applied technique because of its effectiveness in predicting the future events and easy-to-understand characteristics.

Even though decision tree can facilitate decision-makers, the generated tree is normally lengthy because it includes every detail related to each decision choice. A bushy tree is clearly difficult to follow. We thus propose a constrained search method to post-process the induced tree to contain only information of users‟ interest. From the complete decision path for each and every decision choice, our search method can find a smaller set of decision rules that are relevant to users‟ interest. We test our method with the survey information conducted in 2004 by the Public Policy Institute of California and the University of California, Irvine [2]. We select survey details (1,008 records) of the first 30 questions (questions Q02-Q31) as a data set to build a decision tree. Some of these survey questions are as follows:

Q13 First, what do you think is the most important issue facing Orange County today?
Q16 Thinking about the quality of life in Orange County, how do you think things are going- very well, somewhat well, somewhat badly, or very badly?
Q30 Does the cost of your housing place a financial strain on you and your family today?
Q31 Does the cost of your housing make you and your family SERIOUSLY CONSIDER moving away from Orange County?

The complete decision tree contains 165 nodes. We firstly transform the tree into a set of 435 decision rules that users can ask only information of interest. For example, if users focus on the Orange County residents who are seriously thinking about moving out of the county, users can query with the command “findRule([q31-yes])” that means showing rules related to people who are about to move out of the county. The result is reduced to a smaller set of 174 rules. User can further constrain a search mechanism to a query such as “Q31 = yes AND Q16 = somewhat_badly” that means the decision to move away from the county is related to the answer in question 16 that the residents consider their quality of life are going somewhat badly. There are 12 decision rules (as shown in Figure 6) that satisfy this constraint.

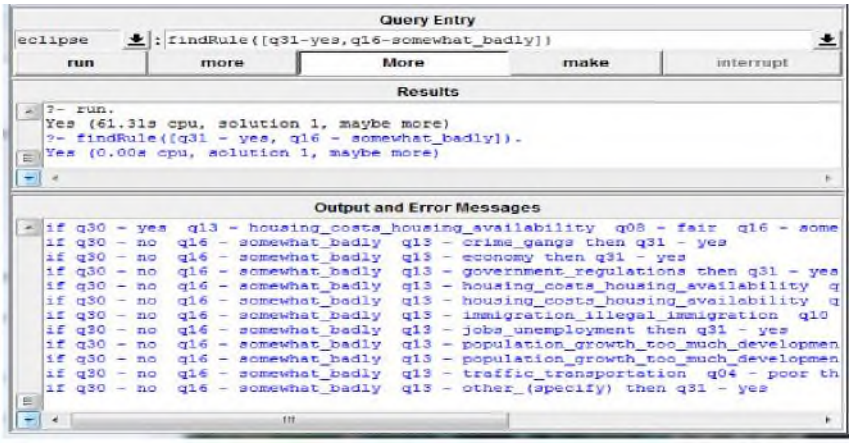

Fig. 6. Search result containing 12 decision rules that comply to the constraints “Q31=yes AND Q16=somewhat_badly”.

6 Conclusion

In the era of digital technologies, most enterprises have collected huge amount of data in an electronic form. Business intelligence (BI) technology has emerged as a tool to support information summarization, pattern extracting, knowledge discovery, and other knowledge-related tasks. The main part of most BI software is the data mining engine to analyze and report relationships that exist in the stored data. Visualization tools are created to help data analysts easily explore the induced information. For extremely large amount of data stored in the data warehouse and data marts, simply explore information and knowledge through the visualize tool is not possible.

We thus propose to put more constraints in the data mining engines. The classification and association mining tasks are further filtered by our constrained search. The mining results can be thus lessen to the comprehensible amount. We plan to extend our research to a more scalable and computational mining engines that can deal with a very large data warehouse.

Acknowledgments. This work has been supported by grants from the National Research Council of Thailand (NRCT) and Suranaree University of Technology.

References

1. Agrawal, R., Srikant, R.: Fast algorithms for mining association rules. In: International Conference on Very Large Data Bases, 487--499 (1994)
2. Baldassare, M.: PPIC Statewide Survey. Public Policy Institute of California (2004)
3. Blake, C. L., Merz, C. J.: Churn data set. UCI Repository of Machine Learning Databases, [http://www.ics.uci.edu/~mlearn/MLRepository.html], University of California, Irvine (1998)
4. Bose, I., Mahapatra, R. K.: Business data mining – a machine learning perspective. Information & Management 39, 211--225 (2001)
5. Chaudhuri, S., Dayal, U., Narasayya, V.: An overview of business intelligence technology. Communication of the ACM 54, 8, 88--98 (2011)
6. Fitriana, R., Eriyatno, Djatna, T.: Progress in business intelligence system research: a literature review. International Journal of Basics & Applied Sciences 11, 3, 96--105 (2011)
7. Isik, O., Jones, M.C., Sidorova, A.: Business intelligence (BI) success and the role of BI capabilities. Intelligent Systems in Accounting, Finance and Management 18, 161--176 (2011)
8. Larose, D. T.: Discovering Knowledge in Data: An Introduction to Data Mining. John Wiley & Sons (2005)
9. Luhn, H.P.: A business intelligence system. IBM Journal of Research and Development 2, 4, 314--319 (1958)
10. Mutanen, T.: Customer churn analysis – a case study. Research Report, No. VTT-R-01184-06. Technical Report Center of Finland (2006)
11. Negash, S.: Business intelligence. Communication of the Association for Information Systems 13, 177--195 (2004)
12. Quinlan, J.R.: Induction of decision tree. Machine Learning 1, 81--106 (1986)
13. Syam, N. B., Hess, J. D.: Acquisition versus retention: competitive customer relationship management. Working Paper, University of Houston, Houston, Texas, U.S.A. (2006)