# Mining Interesting Knowledge Using DM-II

Bing Liu, Wynne Hsu, Yiming Ma and Shu Chen

School of Computing
National University of Singapore
Lower Kent Ridge Road, Singapore 119260
{liub, whsu, maym, chens}@comp.nus.edu.sg

## 1. Introduction

Data mining aims to develop a new generation of tools to intelligently assist humans in analyzing mountains of data. Over the past few years, great progress has been made in both research and applications of data mining. Data mining systems have helped many businesses by exposing previously unknown patterns in their databases, which were used to improve profits, enhance customer services, and ultimately achieve a competitive advantage.

In this paper, we present our unique data mining system DM-II (Data Mining – Integration and Interestingness). DM-II is a PC-based system working in Windows 95/98/NT environment. Apart from the normal components of a data mining system, DM-II has a number of unique and advanced sub-systems. These sub-systems have been applied in many real-life applications, including education applications, insurance application, accident application, disease application, drug screening application, image classification, etc. Here, we focus on discussing the following sub-systems:

**CBA:** CBA originally stands for *Classification-Based on Associations* [10]. It has now been extended with a number of other advanced features. In all, CBA has the following capabilities:

*Building classifiers using association rules*: Traditionally, association rule mining and classifier building are regarded as two different data mining tasks. CBA unifies the two tasks. It is able to generate association rules and builds a classifier using a special subset of the association rules. What is significant is that CBA, in general, produces more accurate classifiers compared to the state-of-the-art classification system C4.5 [16]. It also helps to solve some outstanding problems with the existing classification systems.

*Pruning and summarizing the discovered associations*: One major problem with association rule mining is the huge number of association rules generated. Many of the rules are redundant. Their existence may simply be due to chance rather than true correlation. Thus, pruning should be performed to remove those spurious and insignificant rules. However, the number of rules left after pruning can still be very large. Thus, we define and extract a small subset of the rules, called *direction setting rules* (or DS rules), to summarize the unpruned rules. This is analogous to the summary of a text article that provides the essence of the article. If we are interested in the details of a particular aspect, the summary can point us to it in the article. In the same way, the DS rules give the essence of the domain and points the user to those related details.

*Mining association rules with multiple minimum supports*: The classic association rule model allows only one minimum support for the whole data set. A major shortcoming is that it cannot capture the inherent natures and/or frequency differences of the items in the database. As a result, rules involving those rare but important items are difficult to find. This is called the *rare item problem*. CBA offers an approach to solve this problem.

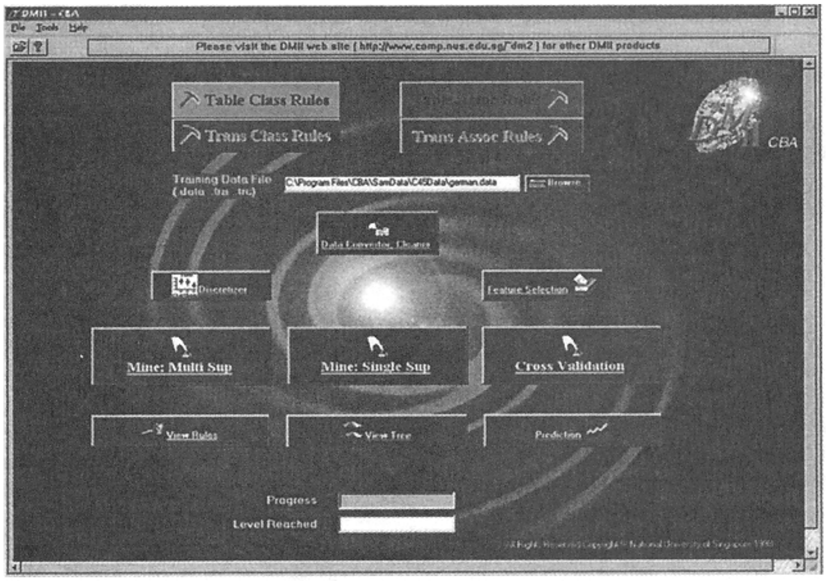
**IAS** (Interestingness Analysis System): One major strength of CBA is that it finds all rules that exist in data. Hence, it can reveal valuable and unexpected information in the database. This strength, however, comes with a drawback. The number of discovered rules can be huge, which makes manual inspection of the rules an almost impossible task. The interestingness analysis system (IAS) helps the user to interactively identify subjectively interesting rules.

**LMS** (Landscape Mining System): Most existing data mining techniques and systems do not distinguish what exist from what do not exist in the data. Thus, a rule that expresses certain property of a group of data points can contain a large empty space in it. Such a rule does not provide a precise description of the area it covers. LMS implements a novel technique to partition the space into various data regions and empty regions. We call this data mining task the *data landscape mining*.

**CMS** (Change Mining System): In businesses, knowing what is changing is crucial. Significant changes often require immediate actions. CMS mines changes and helps the user detect trends.

Below, we discuss these sub-systems in greater detail.

KDD-99 San Diego CA USA
Copyright ACM 1999 1-58113-143-7/99/08...$5.00



Please visit the DMII web site [ http://www.comp.nus.edu.sg/~dm2 ] for other DMII products

Training Data File (.data .txn .trc) C:\Program Files\CBA\SamData\C45Data\german.data

Figure 1. The main interface screen of CBA

## 2. CBA

CBA is a versatile system that has many unique features. We discuss each of them below. Figure 1 shows the main screen interface of CBA.

### 2.1 Building accurate classifiers using association rules

CBA unifies or integrates classification and association rule mining. It focuses on mining a special subset of association rules, called *class association rules* (CARs), which have a fixed target attribute on the right-hand side of the rules. An existing association rule mining algorithm [1] is adapted to mine CARs that satisfy the user specified minimum support and minimum confidence constraints. Mining CARs is by no means a restriction. By setting the target attribute to dummy, CBA can mine normal association rules without a fixed target from table form data or transaction form data.

CBA also provides two efficient algorithms for building accurate classifiers based on the set of discovered CARs. Experiment results show that the classifier built by CBA is in general more accurate [10] than that produced by the state-of-the-art classification system C4.5 (both c4.5rules and c4.5tree) [16]. In addition, CBA has another major advantage over the existing classification systems. Due to the use of the association rule mining technique, CBA is able to discover all rules that exist in the database. Traditional classification systems aims to produce a small set of rules to form a classifier [16]. This results in many interesting/useful rules not being discovered.

CBA's rule generator and classifier builders have been used in all our applications. It is shown to be superior to traditional classification systems because of its high classification accuracy and its ability to generate rules that cannot be generated by classification systems.

### 2.2 Pruning and summarizing the discovered associations

It is well known that in association rule mining many discovered associations are redundant or minor variations of others. Their existence may simply be due to chance rather than true correlation. It is thus important to prune those spurious and insignificant rules. An example of such a rule is shown below.

Example: We have the following two rules,

R1: Job = yes → Loan = approved
[sup = 60%, conf = 90%]

R2: Job=yes, Credit_history = good → Loan = approved
[sup = 40%, conf = 91%]

If we know R1, then R2 is insignificant because it gives little extra information. Its slightly higher confidence is more likely due to chance, rather than true correlation. In CBA, we measure the significance of a rule using chi-square test ($\chi^2$) for correlation from statistics [14].

Pruning can reduce the number of rules substantially. However, the number of unpruned rules can still be very large. They are still difficult to be analyzed manually by a human user. Here, we define and use a special subset of the rules, called *direction setting rules* (or DS rules), to form a summary of the unpruned rules. Essentially, the DS rules are significant association rules that set the directions for other rules to follow. The direction of a rule is the type of correlation it has, i.e., *positive correlation* or *negative correlation* or *independence*, which is also computed using $\chi^2$ test. Let us see an example.

Example: We have the following discovered rules:

R1: Job = yes → Loan = approved
[sup = 40%, conf = 70%]

R2: Own_house = yes → Loan = approved
[sup = 30%, conf =75%]

$\chi^2$ analysis shows that having a job is positively correlated to the grant of a loan, and owning a house is also positively correlated to obtaining a loan. Then, the following association is not so surprising:

R3: Job = yes, Own_house = yes → Loan = approved
[sup = 20%, conf =90%]

because it intuitively follows R1 and R2. We can use R1 and R2 to provide a summary of the three rules. R1 and R2 are DS rules as they set the direction (positive correlation) that is followed by R3. In real-life data sets, a large number of associations are like R3.

From the example, we see that the DS rules give the essential relationships of the domain. The non-DS rule is not surprising if we already know the DS rules. However, this does not say that non-DS rules are not interesting. Non-DS rules can provide further details about the domain, e.g., the non-DS rule above (R3). If the user is interested in the non-DS rules, the DS rules can point him/her to them because non-DS rules are combinations of DS rules.

Experiment results show that although the number of discovered rules can be huge, the number of DS rules is very small. They can be analyzed manually by the human user to obtain the essential relationships in the data. He/she can then focus his/her attention on those interesting aspects of the relationships, and to see the relevant non-DS rules. This technique makes association rule mining effective and practical for data sets whose items are highly correlated. The user can now obtain a complete picture of the domain without being overwhelmed by a huge number of rules. See [11] for the detailed technique.

### 2.3 Mining association rules with multiple minimum supports

Traditional association rule mining aims to discover all item associations (or rules) in the data that satisfy the user-specified minimum support (minsup) and minimum confidence (minconf) constraints. Since only one minsup is used for the whole database, the model implicitly assumes that all items in the data are of the same nature and/or have similar frequencies in the data. This is, however, seldom the case in real-life applications. In many applications, some items appear very frequently in the data, while others rarely appear. If minsup is set too high, those rules that involve rare items will not be found. To find rules that involve both frequent and rare items, minsup has to be set very low. This may cause combinatorial explosion. This dilemma is called the *rare item problem.*

When confronted with this problem, researchers either split the data into a few blocks according to the frequencies of the items and then mine association rules in each block with a different minsup [7], or group a number of related rare items together into an abstract item so that this abstract item is more frequent [7]. However, either approach is satisfactory.

We argue that using a single minsup for the whole data set is inadequate because it cannot capture the inherent natures and/or frequency differences of the items in the database. We have extended the existing association rule model to allow the user to specify multiple minimum supports to reflect different natures and/or frequencies of items. Specifically, the user can specify a different *minimum item support* for each item. Thus, different rules may need to satisfy different minimum supports depending on what items are in the rules. This new model enables us to achieve our objective of producing rare item rules without causing frequent items to generate too many meaningless rules. An efficient algorithm for mining association rules in the new model has been implemented in CBA (see [12]).

## 3. IAS: Interestingness Analysis System

CBA's rule generator is based on association rule mining. Like a normal association rule miner, it can generate a large number of rules. It is almost impossible for a human user to go through the rules manually in order to find the interesting/useful rules. Automated assistance is thus needed. Apart from the pruning and summarization capability of the CBA system, which mainly applies to class association rules, we also have the IAS system to help the user analyze the subjective interestingness of the discovered rules, normal association rules or class association rules.

Existing research in rule interestingness has shown that whether a rule is interesting or not depends on the user's existing knowledge about the domain and his/her current interests [17, 9, 13, 15]. IAS allows the user to specify his/her existing knowledge about the domain. The system then uses this knowledge to analyze the discovered rules according to various interestingness criteria (see below), and through such analysis to identify those potentially interesting rules. IAS also has a visualization system, which enables the user to visually detect interesting rules easily. From a single screen, he/she is able to obtain a global and yet detailed picture of interesting aspects of the discovered rules.

**The specification language:** IAS has a simple specification language to enable the user to express his/her existing knowledge of different degrees of preciseness, namely, *general impressions*, *imprecise concepts*, and *precise knowledge*. The first two types of knowledge represent the user's vague feelings, while the last type represents his/her precise knowledge of the domain.

**Analyzing the discovered rules using user's existing knowledge**: IAS analyzes the set of discovered rules *A*, by "matching" and ranking the rules in *A* against the user's specifications *U*, and in the process to find different types of potentially interesting rules.

Conforming rules: A discovered rule $A_i \in A$ conforms to a piece of user's knowledge $U_j \in U$ if both the conditional and consequent parts of $A_i$ match $U_j \in U$ well.

*Purpose*: they show the user those discovered rules that conform to or are consistent with his/her existing knowledge fully or partially.

Unexpected consequent rules: A discovered rule $A_i \in A$ has unexpected consequents with respect to a $U_j \in U$ if the conditional part of $A_i$ matches $U_j$ well, but not the consequent part.

*Purpose*: they show the user those discovered rules that

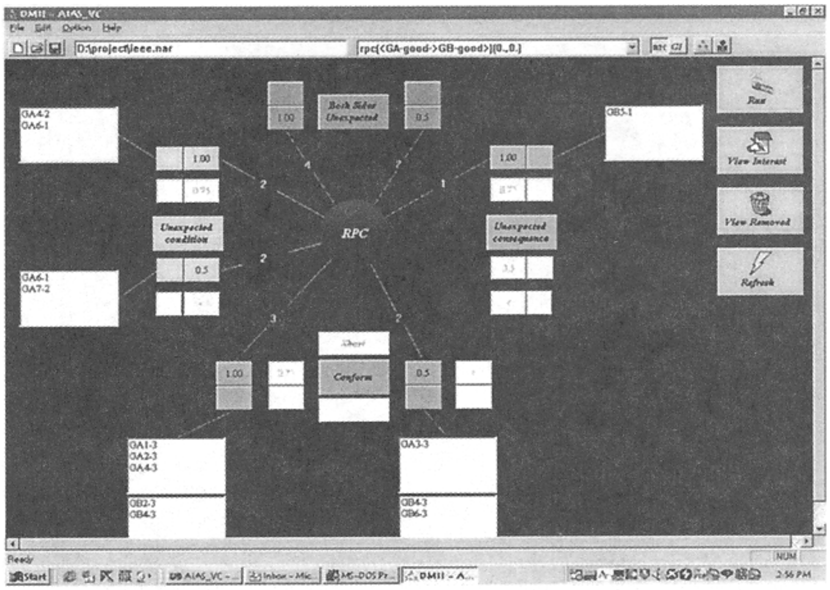


Figure 2. The main visualization screen of IAS

may be contrary to his/her existing knowledge. These rules are often very interesting.

Unexpected condition rules: A discovered rule $A_i \in A$ has unexpected conditions with respect to a $U_j \in U$ if the consequent part of $A_i$ matches $U_j$ well, but not the conditional part.

*Purpose*: these rules show the user that there are other conditions that can lead to the consequent of the specification. The user is thus guided to explore unfamiliar territories.

Both-side unexpected rules: A discovered rule $A_i \in A$ is both-side unexpected with respect to a $U_j \in U$ if both the conditional and consequent parts of the rule $A_i$ do not match $U_j$ well.

*Purpose*: these rules remind the user that there are other rules whose conditions and consequents are not mentioned in his/her specification. It thus helps the user to go beyond his/her existing concept space.

**Visualization component:** After the discovered rules are analyzed, IAS displays different types of potentially interesting rules to the user. It does so by showing the essential aspects of these rules in a single screen such that it can take advantage of the human visual capabilities to identify the truly interesting rules easily and quickly. A screen dump is given in Figure 2.

The visualization component is organized into four units: *conforming rules visualization unit*, *unexpected condition rules visualization unit*, *unexpected consequent rules visualization unit*, *both-side unexpected rules visualization unit*. These units are intuitively located on the screen to facilitate visual inspection.

IAS was used in a number of applications for finding interesting rules, especially the unexpected rules and/or exceptional rules. In these applications, typically there are thousands of discovered rules. Without IAS, it would be very hard to analyze these large numbers of rules. See [13] for the detailed technique employed in IAS.

## 4. LMS: Landscape Mining System

Landscape mining is a new mining task. It is an extension of our work in [8]. We use an example to illustrate the idea of landscape mining. Suppose we have a customer database of 60 records with two numeric attributes, *Age* and *Income*. We know that *Age* ranges from 20 to 80 and *Income* ranges from $40k to $200k per year. Some customers use our service $S_1$ and others use our service $S_2$. These records are plotted in Figure 3(a). We want to characterize the customers using the two services. A rule mining system produces the following three rules, which are also shown in Figure 3(a) as three regions (marked 1, 2 and 3).

1. $Income > 160\text{k} \rightarrow S_1$
2. $Income \leq 160\text{k}, Age \leq 40 \rightarrow S_1$
3. $Income \leq 160\text{k}, Age > 40 \rightarrow S_2$

The 3 rules generalize the data quite well. However, the area covered by each rule contains a large amount of empty space. The rules do not give a precise description of the data because they fail to disclose the fact that there are large areas of non-users (represented by the empty regions). To obtain a *precise description* of the domain, a mining system should generate rules that partition the data as shown in Figure 3(b). It gives us a precise profile of the existing customers (data regions) as well as helps us to identify the potential customers (empty regions). Landscape mining thus enables us to:

1. know precisely the characteristics of our customers. This allows us to provide better services to these existing customers.
2. know the characteristics of those who are not our customers, e.g., $Age \leq 30$, or $Age > 57$. This allows us to probe into the possibilities of modifying the services and/or of doing more promotion in order to attract these

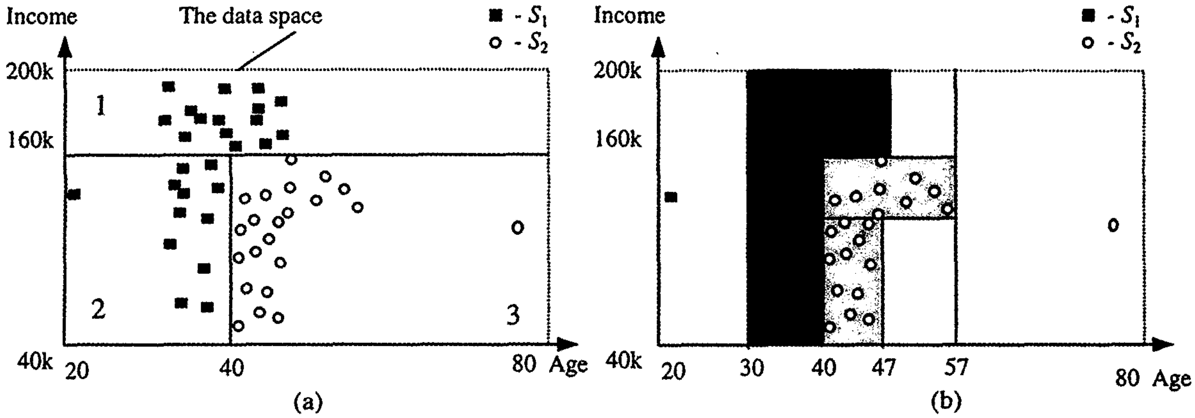


Figure 3: Data landscape mining

potential customers.

A normal rule miner (classification rule miner or association rule miner) cannot give these kinds of crucial and actionable information. LMS implemented a novel technique to perform the landscape mining task.

## 5. CMS: Change Mining System

The world around us changes constantly. Knowing and adapting to changes is an important aspect of our lives. To businesses, knowing what is changing and how it has changed are also of critical importance. There are two main objectives for mining changes:

1. To follow the trends, i.e., to produce products (or services) that suit the changing needs of the people.
2. To remedy or to prevent undesirable changes.

In many applications, mining for changes can be more important than producing models for prediction. A model, no matter how accurate, is passive because it can only predict based on patterns mined in the old data. It should not lead to actions that may change the environment because otherwise, the model will cease to be accurate. Significant changes often trigger immediate actions to alter the domain environment.

Given the data sets collected over time, CMS is able to discover changes that have occurred in the context of decision tree classification and association rule mining.

## 6. Conclusion

In the past few years, many data mining systems, both commercial systems and research prototypes have appeared [e.g., 3, 2, 4, 6]. However, to the best of our knowledge there is no system that is able to unify classification and association rule mining tasks. There is no system that is able to summarize the discovered rules. There is no association rule miner that can mine association rules using multiple minimum supports. There is also no existing system that is able to help the user perform interestingness analysis of the rules. Apart from these unique and advanced features, DM-II also has sub-systems for change mining and for discovering the landscape of data, which are also useful and important for real-life data mining applications.

**Acknowledgement:** We would like to thank many people, Huan Liu, Heng-Siew Han, Minqing Hu, Robin Ng, Ken Wong and Yiyuan Xia, for their contributions to the project. The project is funded by National Science and Technology Board (NSTB) and National University of Singapore (NUS).

## References

[1] Agrawal, R. and Srikant, R. "Fast algorithms for mining association rules." *VLDB-94*, 1994.

[2] Agrawal, R. Mehta, M., Shafer, J. and and Srikant, R. "The Quest data mining system." *KDD-96*, 1996.

[3] Elder, J. F., and Abbott, D. M. A comparison of leading data mining tools. *KDD-98* tutorial.

[4] Han, J., Fu, Y., Wang, W., Chiang, J., Gong, W. Koperski, K. Li, D., Lu, Y. Rajan, A., Stefanovic, N. Xia, B., Zaiane, O. "DBMiner: A system for mining knowledge in large relational databases." *KDD-96*, 1996.

[5] Klemetinen, M., Mannila, H., Ronkainen, P., Toivonen, H., and Verkamo, A.I. "Finding interesting rules from large sets of discovered association rules." *CIKM-94*, 1994.

[6] Kohavi, R. Sommerfield, D., and Dougherty, J. "Data mining using MLC++ a machine learning library in C++." *Processings of tools with AI (TAI-96)*, 1996, pp. 234-245.

[7] Lee, W., Stolfo, S. J., and Mok, K. W. "Mining audit data to build intrusion detection models." *KDD-98*.

[8] Liu, B., Ku, L. P., and Hsu, W. "Discovering Interesting Holes in Data." *IJCAI-97*, pp. 930-935.

[9] Liu, B., Hsu, W. "Post analysis of learned rules." *AAAI-96, 1996*.

[10] Liu, B., Hsu, W. and Ma, Y. M. "Integrating classification and association rule mining." *KDD-98*, 1998, pp. 80-86.

[11] Liu, B. Hsu, W. and Ma, Y. "Pruning and summarizing the discovered associations", *KDD-99*.

[12] Liu, B. Hsu, W. and Ma, Y. "Mining association rules with multiple minimum supports", *KDD-99*.

[13] Liu, B., Hsu, W., Wang, K, and Chen, S. "Visually added exploration of interesting association rules," *PAKDD-99*.

[14] Mills, F. *Statistical Methods*, Pitman, 1955.

[15] Padmanabhan, B., and Tuzhilin, A. "A belief-driven method for discovering unexpected patterns." *KDD-98*, 1998.

[16] Quinlan, J. R. *C4.5: program for machine learning*. Morgan Kaufmann, 1992.

[17] Silberschatz, A., and Tuzhilin, A. "What makes patterns interesting in knowledge discovery systems." *IEEE Trans. on Know. And Data Eng.* 8(6), 1996, pp. 970-974.