

Opportunity Map: A Visualization Framework for Fast Identification of Actionable Knowledge¹

Kaidi Zhao, Bing Liu
Department of Computer Science
University of Illinois at Chicago
851 S. Morgan Street
Chicago, IL 60607. USA
{kzhao, liub}@cs.uic.edu

Thomas M. Tirpak, Weimin Xiao
Motorola Labs
1301 E. Algonquin Rd.
Schaumburg, IL 60196. USA
{T.Tirpak, awx003}@motorola.com

ABSTRACT

Data mining techniques frequently find a large number of patterns or rules, which make it very difficult for a human analyst to interpret the results and to find the truly interesting and actionable rules. Due to the subjective nature of "interestingness", human involvement in the analysis process is crucial. In this paper, we propose a novel visual data mining framework for the purpose of identifying actionable knowledge quickly and easily from discovered rules and data. This framework is called the *Opportunity Map*. It is inspired by some interesting ideas from Quality Engineering, in particular Quality Function Deployment (QFD) and the House of Quality. It associates summarized data or discovered rules with the application objective using an interactive matrix, which enables the user to quickly identify where the opportunities are. The proposed system can be used to visually analyze discovered rules, and other statistical properties of the data. The user can also interactively group actionable attributes and values, and see how they affect the targets of interest. Combined with drill-down and comparative analysis, the user can analyze rules and data at different levels of detail. The proposed visualization framework thus represents a systematic and yet flexible method of rule analysis. Applications of the system to large-scale data sets from our industrial partner have yielded promising results.

Categories and Subject Descriptors

H.2.8 [Information Systems]: Data Management -- Data Mining;
I.3.m [Computer Graphics]: Miscellaneous – Visualization.

General Terms: Management, Design, Human Factors.

Keywords: Information visualization, interactive data exploration, actionable rules, patterns, visual data mining.

1. INTRODUCTION

The objective of many data mining applications is to find useful patterns or rules that can be used in practical tasks. Research to-date has produced many algorithms for pattern or rule mining, e.g., classification rule mining [23] and association rule mining

[2]. These techniques, however, typically generate a large number of rules, most of which are of no use [14][26][22]. A number of techniques have been proposed to help the user find interesting rules [1][3][10][11][14][21][22][29]. The issue of rule interestingness has been studied from two perspectives: objective interestingness and subjective interestingness [14][22][26]. Objective interestingness measures a rule according to its structure and statistical properties, e.g., support, confidence, etc. Subjective interestingness measures a rule according to its interestingness to particular users and applications. Characterizing rules with objective interestingness measures is not sufficient, because many statistically significant rules may not be useful for an application [14][26]. The reason is that the rules may already be known to the user, not related to the user's application, or not actionable, i.e., nothing can be done with the rules. Clearly, subjective interestingness must be considered. Unexpectedness and actionability are two measures of subjective interestingness. Most research to-date has been focused on unexpectedness, as it essentially involves a comparison of discovered rules with the user's existing knowledge, which can often be expressed as rules. Although unexpected rules may help the user to better understand the domain, they may not be actionable for a practical application [1][22]. In practice, actionability is the key. However, actionability is an elusive concept [1][26]. It depends on the task that the user wants to perform, and this aspect is often hard to represent and to relate to discovered rules. So far, limited research has been done with this measure [1][17][22].

The field of Management Science offers some significant insights into the actionability of knowledge. There are well established methodologies and business practices on how to satisfy customer requirements in product design and manufacturing. Quality Function Deployment (QFD) is a management framework and a systematic process for motivating a business to focus on customer needs [6]. It represents a set of product development tools that were developed in Japan to transfer the concepts of quality control from the manufacturing process into the new product development process. The main feature of QFD is to meet market needs by using actual statements from the customer (referred to as the "Voice of the Customer"), technical requirements from the company, and a comprehensive matrix (called the "House of Quality") to identify important activities and prioritize them, and to document information and decisions.

In this paper, we propose a visual data mining framework for fast identification of actionable knowledge. It is inspired by the House

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

CIKM '05, October 31–November 4, 2005, Bremen, Germany.

Copyright 2005 ACM 1-59593-140-6/05/0010...\$5.00.

1. Parts of the work are under patent applications. For most recent advances please contact the authors.

of Quality. The visualization displays a matrix of rich information, which enables the user to easily and quickly focus on opportunities, i.e., potentially actionable knowledge. This framework fits the following general scenario: The user wants to find (actionable) rules or patterns that can be used to solve his/her application problems. Typically in such an application, each data record has a class label. Some classes are important as they represent important real-life problems, e.g., product problems or issues that need to be addressed. In these applications, the user is not interested in prediction or classification, but rather in rules or patterns that can be used to deal with existing issues. In this framework, the identification of actionable rules is regarded as a post-processing step; it assumes that rules have been generated. The framework can also be used to guide the generation of rules in comparative study, as we will explain later in this paper.

The concepts in our data mining setting can be readily mapped to concepts in the House of Quality. Customer requirements correspond to the target classes that the user is interested in. A company's technical requirements in the House of Quality can be mapped to the set of attributes in the data. The combination of the above two concepts forms a two-dimensional matrix. Each cell in the matrix relates a class value to a particular attribute. The relationships between classes and attributes can be expressed in various forms, e.g., rules, distributions of data with regard to certain attributes and classes, etc. In the proposed framework, we are mainly concerned with rules. Our visualization enables the user to quickly determine important classes and actionable attributes and values, and to identify opportunities, i.e., actionable rules that can help solve his/her problems. The Opportunity Map also allows the user to focus on classes and the values of a particular attribute. A novel method for comparative study of rules from different subsets of data is also proposed, thereby enabling the user to perform more detailed studies resulting in more interesting and actionable knowledge.

The proposed visual data mining with Opportunity Map involves:

1. Mining rules from the data, from where we get rules, classes and attributes.
2. Visualizing the rules using the Opportunity Map.
3. Based on the user's application and focus, arranging the map according to class importance and actionability of attributes. This isolates a small area in the matrix/map that may contain actionable rules.
4. Identifying interesting spots in the specific area, i.e., attributes and cells in the matrix/map that are interesting.
5. Drilling down to a particular attribute and classes to find more specific rules. This uses another map of the same nature.
6. Performing comparative study, if needed. In the comparative study, rules mined from different subsets (classes/values) of the data are compared visually to see the differences. This turns out to be very useful in our application.

All these steps can be performed in an iterative manner. Actionable rules are often identified in Steps 5 and 6.

Our work makes the following contributions:

1. To our knowledge, this is the first attempt to study rule actionability using a visualization approach.

2. Ideas from Quality Function Deployment in management science are adapted to rule analysis, which is shown to be very effective. Our framework thus bridges the Management Science and data mining, which makes data mining results more attractive to management teams.
3. It not only enables flexible and interactive analysis of discovered rules, but also compares them with those not discovered rules. This turns out to be very useful in our application because the user frequently wants to see the differences in support and confidences.

Our proposed visual data mining system, Opportunity Map, displays a rich set of information, enabling the user to identify potential opportunities easily. We have used the system in applications with real-life and large-scale datasets from our industrial partner Motorola. The results show that the system is very promising. It helps to find truly interesting and actionable rules from a large number of generated rules. Due to the success, Motorola's technology transfer team is working to make Opportunity Map a tool for product reviews and ongoing decision-making by new product designers and managers.

2. Related Work

Our research is related to three main areas of data mining: interestingness, rule query, and visualization. Below, we review the existing literature in these areas and compare it with our work.

Rule interestingness analysis aims to evaluate the interestingness of discovered rules. As we have discussed in the introduction section, little work has been done on finding actionable rules. Opportunity Map aims to perform the task using an interactive visualization system.

[22] proposes to build an expert system to find actionable rules for an application. However, building an expert system is a major undertaking, which is often more difficult than data mining itself. [1] proposes to organize all actions that the user can perform in a hierarchy. He/she then identifies actionable rules for each node (or action) in the hierarchy. However, finding actionable rules is actually the difficult part of the process as our application shows. Our proposed method aims to help the user find such rules. [17] removes some non-actionable rules, which is an objective analysis of interestingness. It does not find actionable rules for the user.

Another type of data mining techniques allows the user to issue queries to a rule query system to retrieve rules that he/she wants to see [8][19][30]. However, one cannot issue a query to a system to find actionable rules if the user does not already know what actionable rules are. Opportunity Map helps the user to interactively identify actionable knowledge quickly.

Regarding visualization, there are two general approaches, which differ in terms of what content is visualized and manipulated [13]. The first one is raw data visualization, and the second is data mining results visualization. Our proposed work is more related to the latter. For rule visualization, [9] proposes interactive mosaic plots to visualize the contingency tables of the association rules. In [7], classification rules are visualized using rule polygons. Each rule is displayed as a strip covering the area that encompasses its attribute value ranges. [32] introduces a method to visualize association rules without a fixed target. Its rules come from texts collections, and thus it visualizes how words are

related. [33] visualizes the behavior of rules, i.e., changes of supports and confidences over time. In [5], parallel coordinates are used to visualize rules. These methods differ from our approach in terms of both the goal and the visualization.

In [4], a 3D visualization is proposed to visualize association rules by emphasizing their supports and confidences. Although support and confidence are important, they are not the key in finding actionable rules. In [12], a post-processing environment is proposed to browse and visualize association rules. It provides a set of operations to divide a large rule set into smaller ones. The visualization is focused on one subset at a time. Histograms are used to plot the support and confidence values. In [20], important rules in terms of support and confidence values are highlighted with a grid view. However, it does not actively help the user find actionable knowledge.

In [18], ordering of categorical data is studied in order to improve the visualization. The purpose is to have less visualization clutter. It is mainly useful for parallel coordinates [33][34] and other general spreadsheet types of visualization. It is clearly different from our work.

All of the above approaches differ from our proposed Opportunity Map. It is also important to note that Opportunity Map is not just a rule visualization system, but a framework and a process for fast identification of actionable knowledge. It is also able to perform rule comparative analysis, which turns out to be very useful in our application.

3. OPPORTUNITY MAP

This section presents the proposed Opportunity Map framework. We first define the problem that it aims to solve, and then give a short introduction to the House of Quality from Quality Engineering research and practice. After that, we present the Opportunity Map in details.

3.1 Problem Statement

As indicated in the Introduction Section, Opportunity Map is designed for the following type of applications:

1. The dataset has n attributes, A_1, A_2, \dots, A_n . There is also a special *class attribute* C , which has a set of discrete values, C_1, C_2, \dots, C_m . These values are often called *classes*.
2. The classes represent possible states of the system, e.g., various normal states and abnormal states. A subset of the states is interesting or important to the user, and the remaining states are not.
3. The user is interested in finding actionable rules that can help him/her deal with practical problems related to the important subset of states or classes, e.g., the abnormal states/classes.

Clearly, classification/prediction is not the main objective here. Decision trees, naïve Bayesian classification (NB), and SVM are less suitable for the task. Both SVM and NB do not produce interpretable rules. Decision trees only find a small subset of rules, which are often long and hard to understand and to use, and may miss many actionable rules. Thus, they are less effective for explaining the relationships between attributes and classes. The aim here is to find rules that can help solve real-world problems. For example, a New Product Introduction Engineer may want to find the relationships between product attributes and abnormal

classes in records of product performance data, thereby providing insights regarding how to modify the product design and reduce the abnormal cases occurring in future performance data.

Since the applications addressed here include a target attribute (or a class attribute) representing product problems, we mine rules using a class association rule miner, CBA [15]. A *class association rule* is an association rule with only a class value on the right-hand-side of the rule. We also use multiple minimum supports in rule mining because the classes are extremely unbalanced. Abnormal cases are very rare and some abnormal cases are also rarer than others. The idea of mining association rules with multiple minimum supports can be found in [16].

3.2 The House of Quality

Quality Function Deployment (QFD) is a widely used Industrial Engineering technique for systematically focusing on customers' requirements. Usually, cross-functional teams within the business use it to identify and resolve issues involved in product development, process control, services, etc [6]. Here, we introduce its first step, which is called the "House of Quality". It organizes important activities and information (Figure 1).

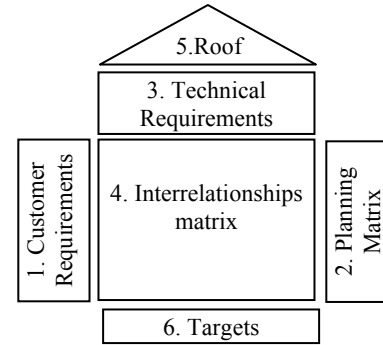


Figure 1. The House of Quality

The six components are:

1. Customer Requirements: A structured list of customer requirements for a product, usually described in their words (also called Voice of Customer).
2. Planning Matrix: A matrix quantifying the customers' requirement priorities and their perceptions of the performance of existing products. These priorities can be adjusted based on the issues that the design team identifies.
3. Technical Requirements: A set of engineering characteristics to meet the customer needs. It describes the product in the terms of the company (thus called the Voice of the Company).
4. Interrelationships matrix: A matrix that relates customer requirements and technical requirements to identify issues, and group these issues into different sectors according to their importance and priorities.
5. Roof: A display of where the technical requirements that characterize the product, support or impede one another.
6. Targets: A summary of the conclusions drawn from the data contained in the entire matrix and the team's discussions.

The House of Quality represents a systematic process and tools for decision making and issue finding [6][28].

In our data mining context, we are primarily interested in the first four components of the House of Quality:

1. Customer Requirements: Application requirements expressed as the classes that are important to the user (or customer).
2. Planning Matrix: A matrix quantifying the user's requirement priorities.
3. Technical Requirements: Attributes and values (other than the class attribute and its class values) that can be used to deal with the problems of certain classes.
4. Interrelationships matrix: Linkages between classes and attributes to identify issues and opportunities for problem solving. This will be discussed in greater detail below.

Our goal is to identify important rules (which are attribute-value combinations) that affect important classes. They form the key opportunities for the user to focus on in order to solve real-world problems. Since the first three components are fairly simple, we focus on component 4.

3.3 Matrix Visualization - Opportunity Map

The Opportunity Map displays a matrix, where the X-axis lists all the attributes, and the Y-axis lists all the classes. Figure 2 shows an example. With the help of cell visualization (CV), the user can quickly identify key attributes (see Section 3.4 also). Priority regions (see below) in the matrix are created by arranging the attributes and classes according to the application. When an interesting attribute is identified, the user can drill down to that attribute for further analysis. We present the details below.

Important Classes: As mentioned earlier, the user is often interested in only a subset of the classes. In matrix visualization, the classes are ranked according to their importance to the user, i.e., the important classes are placed at the upper part of the visualization, while the less important classes are located at the lower part of the matrix.

Actionable Attributes: Attributes are divided into two groups according to their actionability. An attribute is actionable if the user is able to do something with that attribute to achieve some desired effects. An attribute is not actionable if it is out of the user's control. For example, in the medical decision-support systems, a patient's age is not an actionable attribute, because the doctor cannot change the age of the patient. Note that a non-actionable attribute in one application may be actionable in another application.

In Opportunity Map, actionable attributes are placed on the left side of the matrix, while non-actionable attributes are placed on the right.

With this placement of classes and attributes, the matrix visualization is divided into four sectors as shown in Figure 2.

Sector 1 (upper-left): This sector contains important classes and actionable attributes. This is the most important sector and represents the best opportunities. It is thus the priority area on which the user should focus.

Sector 2 (upper-right): Although it is related to important classes, the attributes in this sector are not actionable by the user in this particular application. Rules in this area may, however, help the user to better understand the application domain.

Sector 3 (lower-left): This area contains the less important classes with actionable attributes. The knowledge discovered from this area can be acted upon, but is generally of less interest to the user for the current application.

Sector 4 (lower-right): Rules in this area are not important and not actionable.

It should be noted that the important classes and actionable attributes are domain- and problem-dependent. It is quite possible that rules discovered from a non-actionable area for one application, may be actionable for another application. Thus, we would encourage the users to examine all four sectors of the Opportunity Map.

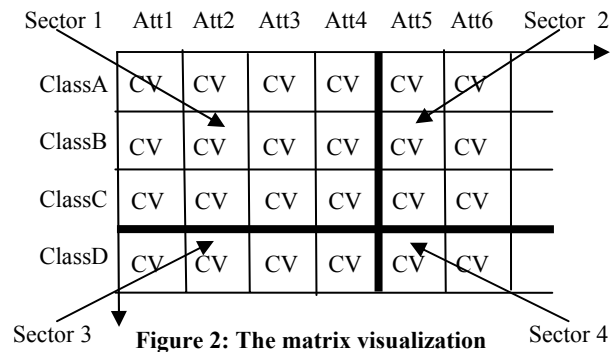


Figure 2: The matrix visualization

After the user finds a certain interesting attribute, he/she can study details by using *drill-down visualization* on that attribute. A drill-down on an attribute allows the user to visualize that attribute's values with respect to all the classes, the rules, and the original data. The visualization is similar to that in Figure 2. Classes are still associated with Y axis, but the X axis shows the values of that drill-down attribute (values are discretized/binned if the attribute takes continuous values). In this way, the user is able to analyze the values of the attribute with respect to the classes, its distribution, etc. Next section will give an example.

In Opportunity Map, if the user drills down on an attribute, a new visualization frame is created, which is displayed on top of the existing matrix. The user can switch among these frames easily.

3.4 Cell Visualization (CV)

We now present the visualization in each cell/grid, which helps the user to decide where to focus on. In our current system, each cell shows two major pieces of information:

Rules: The visualization allows the user to see how many rules are related to an attribute (or a value, if it is in a drill down visualization) and a class, and how strong these rules are in terms of supports and confidences (ranked by confidences).

Data: The visualization shows the data points covered by the rules and the total number of data points in a cell.

Figure 3 shows an example. The figure on the left is the actual visualization, and the figure on the right explains it. The top part of the cell visualization visualizes the set of rules in that cell. First, all the rules are ordered in descending order according to their confidence values. After that, each rule is visualized as a small bar. The height indicates the rule's confidence, and the width indicates its support. Two colors (orange and purple by

default) are used in a round robin fashion so that the user is able to distinguish different rules. Since the human factors research literature has reported that human eyes are more sensitive to the outline of a figure than to its content within that outline, we do not need to show the full height of the bars in order to convey the confidence information. We only draw each bar with a fixed length below the outline (manually highlighted in green in Figure 3). This leaves us a large block of space in the center (and lower) part for communicating other information. By default, the block is blue colored, using the depth of the blue (saturation value) as a rough indication of how many data points there are. The number of rules is written in the middle of the cell. Not all rules are displayed if there are too many of them. This is compensated by allowing the user to dynamically scaling and resizing the grids so that more rules can be visualized.

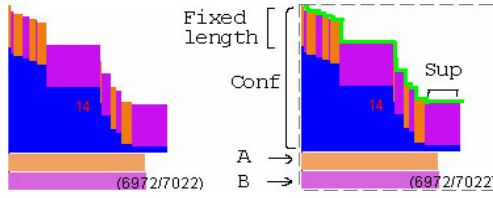


Figure 3. Details of cell visualization

Now, let us summarize some useful properties of the rule visualization in each cell.

- It allows the user to instantly see which cell contains more data based on the prominent depth (saturation) of blue color.
- It also enables the user to compare sets of rules and instantly see which set is a "stronger" set. A set of rules is "stronger" if its rules on average have a higher confidence and support. This can be easily seen from the outline of the rules.
- It clearly shows strong rules, which are wide (high support) and/or tall (high confidence) bars. These rules are likely to be very important for an application.

Brushing and dynamic linking are used in the visualization so that when the user points the mouse to a rule (or a cell), the details of that rule (or cell) will be shown in a separate window on the right. Also, related attributes of the pointed rule will be highlighted in the matrix visualization. Examples will be given in the next section using real world data.

In Figure 3, there are two optional small horizontal bars at the bottom of the cell visualization, labeled as bar A and bar B, as a way to indicate certain values of the underlying data. By default, they (proportional) indicate the number of data points in that grid and the data number covered by the rules in that grid. This gives the user a brief idea of data distribution and the quality of the rules in that grid. For the example, the rules in Figure 3 are good in the sense that they almost cover all the data in that grid since the two bars are of similar length. The exact counts are also optionally shown at the lower right of the grid (14 rules cover 6972 data points out of total 7022 data points there).

3.5 Comparative Study

In our application, it was found that although some individual rules are useful, the knowledge gained through comparative study of subsets of the data can also be very interesting and actionable.

For example, in the product design domain, one may want to compare the rules for two products to find out why one performs better than the other (which can be seen easily from drill-down and cell visualization because product model is an attribute).

A simplistic way of doing this comparative study is to find rules related to product 1 and rules related to product 2, and then show these two rule sets to the user side-by-side. However, this does not work well because rules in the two sets can be quite different due to minimum support and minimum confidence constraints. Thus, we propose the following approach.

Assume that the data subsets of interests are, D_1, D_2, \dots, D_q . For example, D_i represents the set of data records of product i . We perform the following steps:

1. **Mining rules in each data subset:** Let the set of rules mined from data subset D_i be R_i . Two rules from different rule sets are considered the same if they have the same left-hand-side and right-hand-side. The set R of rules that will be compared in the next step is defined as follows:

$$R = \{r \mid r \in (R_1 \cup R_2 \cup \dots \cup R_q)\}$$

To prepare for comparison, a data scan is performed to obtain the support and confidence values of all the rules.

2. **Visualize the rules:** After all the necessary information about supports and confidences of each rule r in R in all data subsets are obtained, we visualize them using a bar-chart based rule visualization (where width is the support and height is the confidence). An example is shown in Section 4 of this paper.

3.6 Visualization Complexity & System Extensibility

A visualization or a visual data mining system is effective only when the complexity of the visualization is within user's common perception. Opportunity Map balances this well without overwhelming the user. Due to the way we design the visualization metaphor, our users confirm these simple methods usually give them easy start and lead them to interesting knowledge discovery: start with prominent cells (with deeper color, larger bars) and examine them. Drilling down on attributes and examining the drill down visualization and rules usually give hints of other related attributes (or cells) to study further. This brings back to the first step but with enriched knowledge and clearer view of the nature of the data.

The Opportunity Map system also provides many general-purpose functionalities and operations. The framework and Cell Visualization are extensible. Correlations and other statistical results can also be visualized depending on the needs. Our subsequent research work will explore in these directions.

4. CASE STUDY

This section addresses the capabilities and evaluation of the proposed system. In general, it is hard to have an objective measure of effectiveness for such a visualization and visual data mining system. It can only be evaluated subjectively by the people who use it for real-life applications. This section presents a case study based on our real-life applications using the Opportunity Map for Motorola's Mobile Devices Business unit. Because of the need to preserve the confidentiality of the

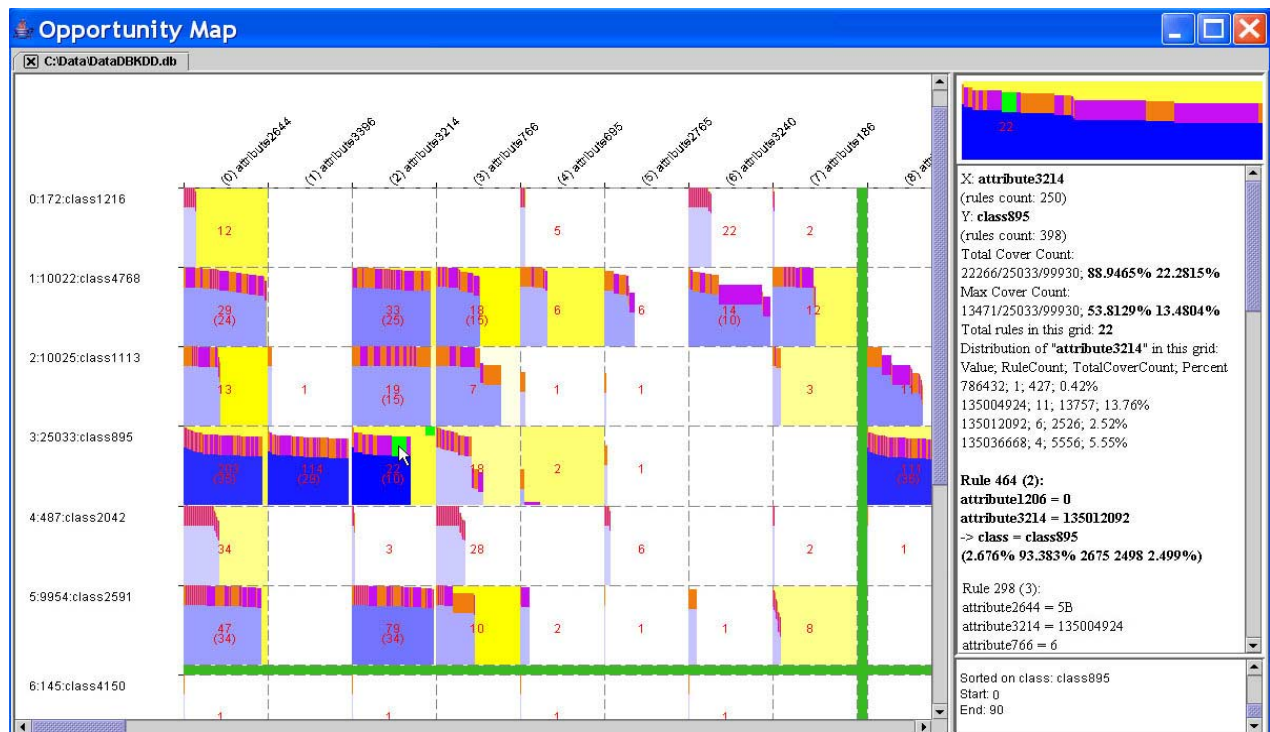


Figure 4. Initial view of Opportunity Map.

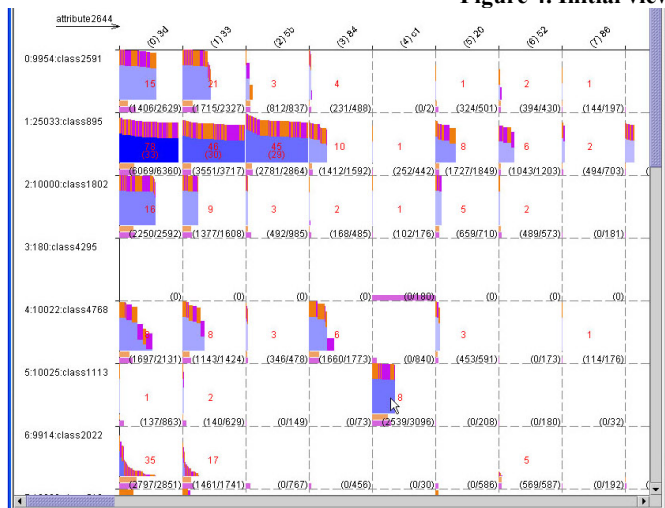


Figure 5. Drill down visualization on one attribute.

application and the actual data, class names and attribute names in our case study are presented with generic names such as "class123" or "attribute456". Some data values are replaced by generic values such as 1, 2, 3, c1, a1, etc.

The Motorola datasets that we used all have more than 500 attributes and more than 100 classes. Based on conversations with human experts, it was learned that there are about fifteen classes that are very important for this application. The classes represent different types of product failure modes that occur. The objective of this data mining application is to find patterns or rules in the data in order to reduce the occurrence of such problems.

The results presented here are based on a dataset with 100,000 records. The classes are highly imbalanced, because product

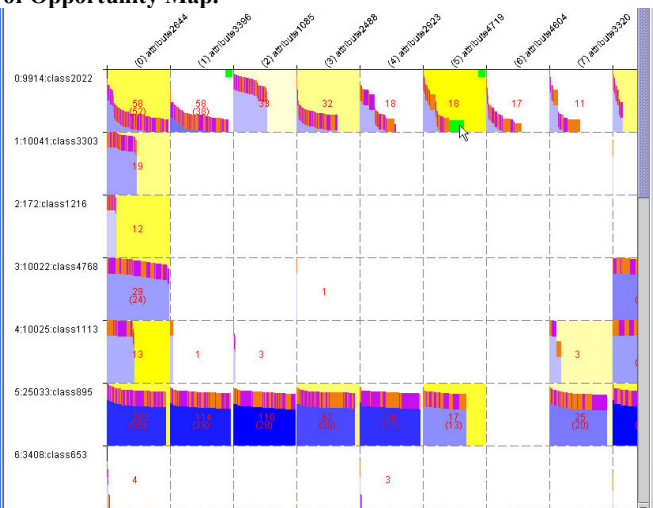


Figure 6. Sorting on class2022.

failure modes are an infrequent phenomenon. We used the class association rule miner CBA [15] to mine rules.

After loading the rules and data, the user will see the first Opportunity Map screen as in Figure 4. The main window (on the left) displays the matrix visualization of Opportunity Map. All the user interaction with the system is performed in this window. The horizontal (X) axis shows all the attributes or variables. The vertical (Y) axis presents the names of the classes. The two thick green lines are used to divide the matrix into four sectors as described in Section 3.3. In the subsequent figures, these two lines are not visible due to the limited page size in this paper (there are many actionable attributes and important classes).

The window on the right is called the *information window*, which

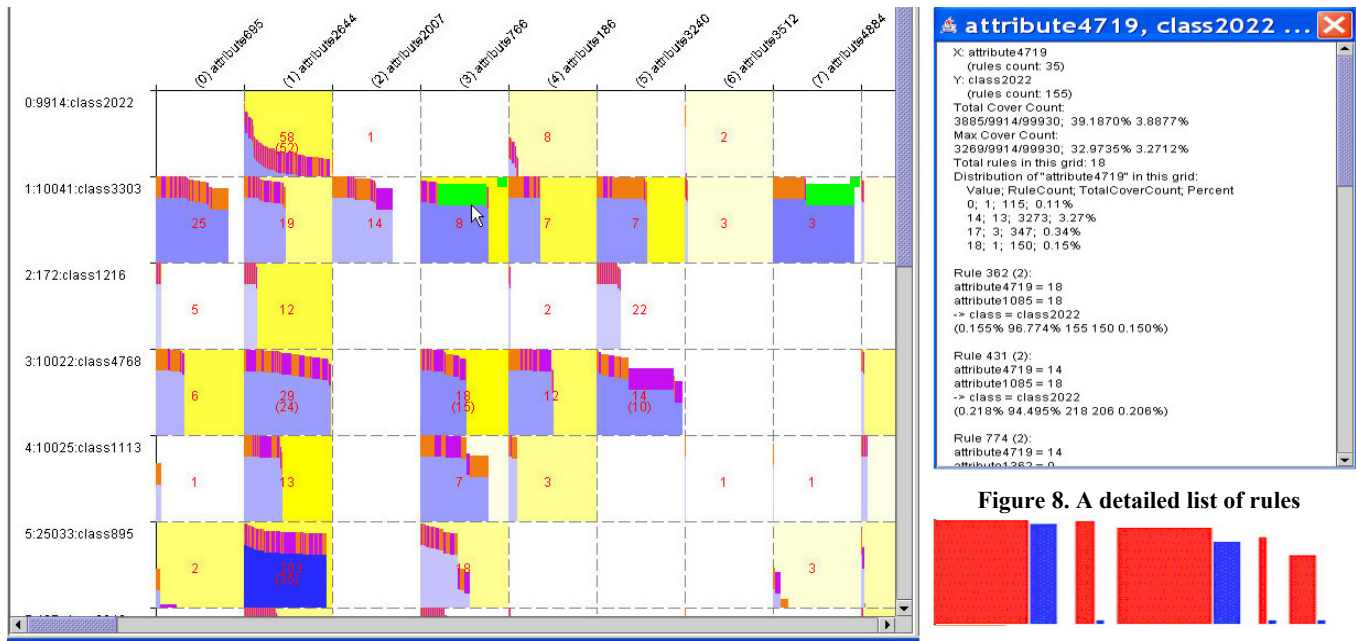


Figure 7. Sorting on class3303

displays the detailed information as the user moves the mouse cursor over the main window. A small log window on the lower-right corner logs the operations performed, and provides the user some hints when applicable.

To start the analysis, the user first puts important classes and actionable attributes in the top-left sector by drag-and-drop. He/she then sorts the attributes according to the rule counts (or any other criteria supported by the system) for the important classes (sorting is done sector by sector only), and examines each attribute by scrolling to the right. The visualization reveals that attribute2644 (left most in Figure 4) plays a very active role in many classes. Drilling down on the attribute may reveal its relations with those classes in detail. Figure 5 gives the drill-down view of attribute2644.

Immediately, from the prominent purple bar in the middle of the visualization (Figure 5), the user is able to identify some very important knowledge. Namely, class4295 (which is also an important class) happens only when attribute2644 takes the value of c1. This leads the user to investigate the possible reasons, using the available knowledge about the application domain. It should be noted that there is no rule discovered for that cell in Figure 5. This is a result of there being not many data points for that class (as was noted earlier, the classes are quite imbalanced). Therefore, it fails to satisfy the minimum support requirement of rule generation. In this way, Opportunity Map successfully helps the user discover this information while rule mining fails to do.

At the same time, the user can see that it is difficult to describe some classes by rules, as these classes require a large collection of rules to cover the corresponding data. One such example in Figure 5 is class895 when attribute2644's value is 3d. Also, it is interesting to see that some data (such as class2591 at value 3d) is hard to summarize with rules. Even with fifteen rules, only 1,406 data points out of the total 2629 data points are covered. When presented with such visualization for this data set, the Motorola engineers started a discussion of the possible relationships for the

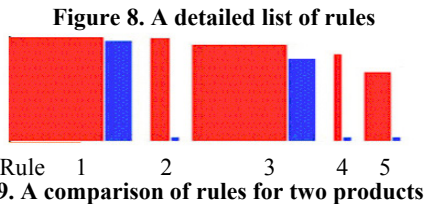


Figure 9. A comparison of rules for two products

specific attributes and data, leading to some very useful insights. For a cell, the user can bring up a detailed rule window to see the exact rules, as shown in Figure 8.

The user then is interested in performing comparison of classes, and wants to find out different dominating attributes and rules for each class. This is made easy by several built-in sorting methods within the Opportunity Map.

Figure 6 shows the results of sorting on class2022. Figure 7 shows the results of sorting on class3303. The two visualizations indicate that these two classes can be characterized by very different attributes, as shown on X axis. When the Motorola engineers were presented with these visualizations, they looked through these attributes and rules one by one, based on their domain knowledge to either interpret the reasons, or formulate new hypotheses, which lead to further detailed investigations (unfortunately, we could not disclose the actions). In Figure 6, it is notable that class2022 and class895 have many common related attributes, except for only a few, e.g., attribute4604. This leads to the hypothesis that the problems represented by these two classes can be dealt with together.

In all the visualization, dynamic linking is provided whenever the user points to a rule. For example, in Figure 7, the green bar highlights those attributes involved in the rule to which the user points. If those attributes are not sorted together, a mouse click can position them adjacent to each other for a better view.

Finally, it was found that in a drill down visualization (similar to that in Figure 5) that one product had many more failures related to a problem class than another product. This immediately leads to a comparative study of the two products as requested by a Motorola engineer. Figure 9 shows this case with rules from the two products. Red is for product 1, and blue is for product 2. Each pair of bars represents one rule, i.e., the confidence (height) and support (width) of the first bar is for product 1, and the confidence and support of the second bar is for product 2. We can clearly see that Rules 1, 2, 3, and 5 indicate more failures for product 1

(higher supports). Rules 2, 4, and 5 indicate that given the same conditions, product 2 is much less likely to result in failure. The insights from these rules are immediately actionable, as engineers can review the entire set, study why these are cases and identify/propose possible design changes for product 2.

In summary, Opportunity Map helps to find many pieces of truly useful and actionable knowledge from a large set of discovered rules. Motorola technology transfer team is working to make the system a tool for their product reviews and ongoing decision-making by new product designers and managers.

5. CONCLUSIONS

In this paper, we proposed the Opportunity Map visualization framework, and described its application for fast identification of actionable knowledge. This framework, which was inspired by the House of Quality from Industrial Engineering, visualizes the user's needs, e.g., problem classes, and attributes that produce rules as a matrix. Opportunity Map clearly shows how attributes and their values are linked to the important classes. Combined with actionable attributes, the user can quickly focus on useful rules (and interesting cells). With sophisticated cell visualization and comparative study, Opportunity Map represents a systematic way of post-analysis of discovered rules with convenient visualization support. In our applications with real-life, large-scale data sets from our industrial partner, it has been possible to find the truly interesting and useful rules and patterns from a large number of rules generated by data mining. Thus, the technology transfer team of our industrial partner is working to make the Opportunity Map a part of product reviews and ongoing decision-making by new product designers and managers.

6. ACKNOWLEDGEMENT

We would like to thank Michael Kramer and Dan DeClerck for many useful discussions and suggestions. Thank you also to Mike Kotzin of Motorola's Mobile Devices Business for supporting this project through the Illinois Manufacturing Research Center.

7. REFERENCES

- [1] Adomavicius, G. and Tuzhilin, A. Discovery of actionable patterns in databases: the action hierarchy approach. *KDD-97*, 1997.
- [2] Agrawal, R. and Srikant, R. Fast algorithms for mining association rules. *VLDB-1994*.
- [3] Bayardo, R. and Agrawal, R. Mining the most interesting rules. *KDD-99*, 1999.
- [4] Blanchard J, Guillet F, Briand H. Exploratory Visualization for Association Rule Rummaging. *KDD-03 Workshop on Multimedia Data Mining (MDM-03)*, 2003.
- [5] Bruzzese D., Davino C. Visual Post-Analysis of Association Rules. *ECML/PKDD02 VDM Workshop*, 2002.
- [6] Cohen L. *Quality Function Deployment*. Prentice Hall, 1995.
- [7] Han J, Cercone N. RuleViz: A Model for Visualizing Knowledge Discovery Process. *KDD-00*, 2000.
- [8] Han, J., Fu, Y., Wang, W., Koperski, K. and Zaijane, O. DMQL: a data mining query language for relational databases. *SIGMOD Workshop on DMKD*, 1996.
- [9] Hofmann H, Siebes A., Wilhelm, A. Visualizing association rules with interactive mosaic plots. *KDD-00*.
- [10] Hilderman, R., Hamilton, H. Evaluation of Interestingness measures for ranking discovered knowledge. *PAKDD-2001*.
- [11] Jaroszewicz, S., and Simovici, D. Interestingness of Frequent Itemsets Using Bayesian Networks as Background Knowledge. *KDD-04*, 2004.
- [12] Jorge A., Pocas J., Azevedo P. Post-processing environment for browsing large sets of association rules. *PKDD-02 VDM Workshop*, 2002.
- [13] Keim, D. Information Visualization and Visual Data Mining. *IEEE Trans. Vis. Comput. Graph*, 8(1): 1-8, 2002.
- [14] Liu, B., Hsu, W., and Chen, S. Using general impressions to analyze discovered classification rules. *KDD-97*, 1997.
- [15] Liu B., Hsu W, Ma Y. Integrating Classification and Association Rule Mining. *KDD-98*, 1998.
- [16] Liu B., Hsu W, Ma Y. Mining Association Rules with Multiple Minimum Supports. *KDD-99*, San Diego, USA.
- [17] Liu, B., Hsu, H., Ma, Y. Identifying Non-Actionable Association Rules. *KDD-01*, 2001.
- [18] Ma S., Hellerstein J. Ordering Categorical Data to Improve Visualization. *INFOVIS-99*, 1999.
- [19] Meo, R. Psaila, G., and Ceri, S. A New SQL-like operator for mining association rules. *VLDB-96*, 1996.
- [20] Ong K-H, Ong K-L, Ng W-K, Lim E-P. CrystalClear: Active Visualization of Association Rules. *ICDM-02 Workshop on Active Mining (AM-02)*, 2002.
- [21] Padmanabhan, B. and Tuzhilin, A. A belief-driven method for discovering unexpected patterns. *KDD-98*, 1998.
- [22] Piatetsky-Shapiro, G., and Matheus, C. The interestingness of deviations. *KDD-94*, 1994.
- [23] Quinlan, J. R. 1992. *C4.5: program for machine learning*. Morgan Kaufmann.
- [24] Rasmussen M, Karypis G. gCLUTO - an interactive clustering, visualization, and analysis system. Technical Report # 04-021, University of Minnesota, 2004.
- [25] Shah, D., Lakshmanan, L.V.S, Ramamritham, K., and Sudarshan, S., 1999. Interestingness and Pruning of Mined Patterns. *DMKD-99*, 1999.
- [26] Silberschatz, A., and Tuzhilin, A. What makes patterns interesting in knowledge discovery systems. *IEEE Trans. on Know. and Data Eng.* 8(6), 1996, 970-974.
- [27] Suzuki, E. Autonomous discovery of reliable exception rules. *KDD-97*, 1997.
- [28] Terninko J. *Step-by-Step QFD: Customer-Driven Product Design*. Saint Lucie Press. 1997.
- [29] Tuzhilin, A. and Adomavicius, G. Handling Very Large Numbers of Association Rules in the Analysis of Microarray Data. *KDD-02*, 2002.
- [30] Tuzhilin, A., and Liu, B. Querying multiple sets of discovered rules. *KDD-2002*, 2002.
- [31] Virmani A., Imielinski, T. M-SQL: A query language for database mining. *Journal of DMKD*, 1999.
- [32] Wong P-C, Whitney P, Thomas J. Visualizing Association Rules for Text Mining. *INFOVIS-99*, 1999.
- [33] Zhao K. Liu B. Visual Analysis of the Behavior of Discovered Rules. *KDD-2001 Workshop on Visual DM*.
- [34] Zhao, K. Liu, B., Tirpak, T., and Schaller, A. V-Miner: Using Enhanced Parallel Coordinates to Mine Product Design and Test Data. *KDD-2004*.