A Post-processing Environment for Browsing Large Sets of Association Rules [1]

Alipio Jorge[1] , João Poças[2], Paulo Azevedo[3]

[1] LIACC/FEP, Universidade do Porto, Portugal
Rua do Campo Alegre, 823
4150-180 Porto, Portugal
+351 22 607 88 30 (office), +351 93 424 07 19 (private)
amjorge@liacc.up.pt
[2] Instituto Nacional de Estatística, Portugal
joao.pocas@ine.pt
[3] Universidade do Minho, Portugal
pja@di.uminho.pt

Abstract. Association rule engines typically output a very large set of rules. Despite the fact that association rules are regarded as highly comprehensible and useful for data mining and decision support in fields such as marketing, retail, medicine, demographics, among others, lengthy outputs may discourage users from using the technique. In this paper we propose a post-processing methodology and tool for browsing/ visualizing large sets of association rules. The method is based on a set of operators that transform sets of rules into sets of rules, allowing focusing on interesting regions of the rule space. Each set of rules can be then depicted with different graphical representations. The tool is web-based and uses SVG. The input set of association rules is given in PMML.

Keywords: Data mining, association rules, post processing, decision support, visualization.

Paper track.

DECLARATION: A version of this paper has been submitted to another conference.

[1] This work is supported by the European Union grant IST-1999-11.495 Sol-Eu-Net and the POSI/2001/Class Project sponsored by FCT

A Post-processing Environment for Browsing Large Sets of Association Rules [2]

Alipio Jorge[1] , João Poças[2], Paulo Azevedo[3]

[1] LIACC/FEP, Universidade do Porto, Portugal
amjorge@liacc.up.pt
[2] Instituto Nacional de Estatística, Portugal
joao.pocas@ine.pt
[3] Universidade do Minho, Portugal
pja@di.uminho.pt

Abstract. Association rule engines typically output a very large set of rules. Despite the fact that association rules are regarded as highly comprehensible and useful for data mining and decision support in fields such as marketing, retail, medicine, demographics, among others, lengthy outputs may discourage users from using the technique. In this paper we propose a post-processing methodology and tool for browsing/ visualizing large sets of association rules. The method is based on a set of operators that transform sets of rules into sets of rules, allowing focusing on interesting regions of the rule space. Each set of rules can be then depicted with different graphical representations. The tool is web-based and uses SVG. The input set of association rules is given in PMML.

1 Introduction

Association Rule (AR) discovery (Agrawal et al. 96) is many times used, for decision support, in data mining applications like market basket analysis, marketing, retail, study of census data, analysis of medical data, among others. This type of knowledge discovery is adequate when the data mining task has no single concrete objective to fulfil (such as how to discriminate good clients from bad ones), as it is the case in classification or regression. Instead, the use of AR allows the decision maker/ knowledge seeker to have many different views on the data. There may be a set of general goals possibly not measurable (like "what characterizes a good client?", "which important groups of clients do I have?", "which products do which clients typically buy?"). Moreover, the decision maker may even find relevant patterns that do not correspond to any question formulated beforehand. This style of data mining is sometimes called "fishing" (for knowledge).

[2] This work is supported by the European Union grant IST-1999-11.495 Sol-Eu-Net and the POSI/2001/Class Project sponsored by FCT

Due to the data characterization objectives of the association rule discovery task, AR discovery algorithms produce a complete set of rules above user-provided thresholds (typically minimal support and minimal confidence, defined in Section 2). This implies that the output of such an algorithm is a very large set of rules, which can easily get to the thousands, overwhelming the user. To make things worse, the typical association rule algorithm outputs the list of rules as a long text (even in the case of commercial tools like SPSS Clementine), and lacks post processing (sometimes also called rule mining) facilities for inspecting the set of produced rules.

In this paper we propose a method and tool for the browsing and visualization of association rules. The tool reads sets of rules represented in the proposed standard for predictive models, PMML (Data Mining Group). The complete set of rules can then be browsed by applying rule set operators based on the generality relation between itemsets. The set of rules resulting from each operation can be viewed as a list or can be graphically summarized through a number of techniques.

This paper is organized as follows: we start by introducing the basic notions related to association rule discovery, and association rule space. We then describe PEAR, the post processing environment for association rules and its implementation. We describe the set of operators in more detail, show one example of the application of PEAR, compare with related work and conclude, also suggesting the next steps of our work.

2 Association Rules

An association rule $A \rightarrow B$ represents a relationship between the sets of items A and B. Each item I is an atom representing the presence of a particular object. The relation is characterized by two measures: support and confidence of the rule. The support of a rule R within a dataset D, where D itself is a collection of sets of items (or itemsets), is the number of transactions in D that contain all the elements in $A \cup B$. The confidence of the rule is the proportion of transactions that contain $A \cup B$ with respect to the number of transactions that contain A. Each rule represents a pattern captured in a dataset. The support of the rule is the commonness of that pattern, while the confidence measures its predictive ability.

The most common algorithm for discovering AR from a dataset D is APRIORI (Agrawal et al. 96). This algorithm produces all the association rules that can be found from a dataset D above given values of support and confidence, usually referred to as *minsup* and *minconf*. APRIORI has many variants with more appealing computational properties, but that produce exactly the same set of rules.

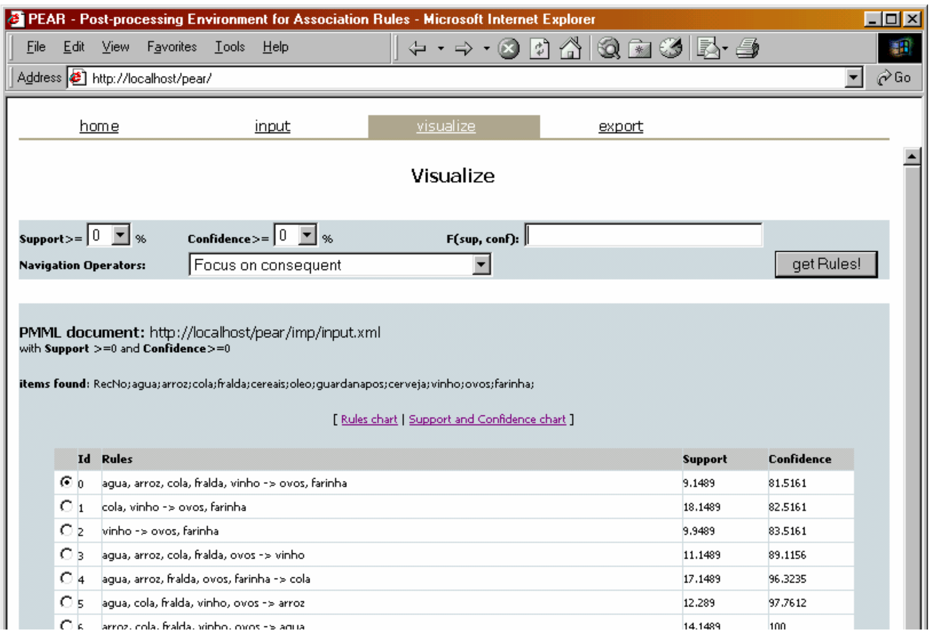
2.1 The Association Rule space

The space of itemsets I can be structured in a lattice with the $\subseteq$ relation between sets. The empty itemset $\varnothing$ is at the bottom of the lattice and the set of all itemsets at the top. The $\subseteq$ relation also corresponds to the generality relation between itemsets.

To structure the set of rules, we need a number of lattices, corresponding each lattice to one particular itemset that appears as the antecedent, or to one itemset that occurs as a consequent. For example, the rule $\{a,b,c\}\rightarrow\{d,e\}$, belongs to two lattices: the one of the rules with antecedent $\{a,b,c\}$, structured by the generality relation over the consequent, and the lattice of rules with $\{d,e\}$ as a consequent, structured by the generality relation over the antecedents of the rules.

We can view this collection of lattices as a grid, where each rule belongs to one intersection of two lattices. The idea behind the rule browsing approach we present, is that the user can visit one of these lattices (or part of it) at a time, and take one particular intersection to move into another lattice (set of rules).

3 PEAR: a web-based AR browser


PEAR - Post-processing Environment for Association Rules - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Address http://localhost/pear/
Go
home
input
visualize
export
Visualize
get Rules!

Support>= 0 % **Confidence>=** 0 % **F(sup, conf):**

Navigation Operators: Focus on consequent

PMML document: http://localhost/pear/imp/input.xml
with **Support** >=0 and **Confidence**>=0

items found: RecNo;agua;arroz;cola;fralda;cereais;oleo;guardanapos;cerveja;vinho;ovos;farinha;

[Rules chart | Support and Confidence chart]

	Id	Rules	Support	Confidence
●	0	agua, arroz, cola, fralda, vinho -> ovos, farinha	9.1489	81.5161
○	1	cola, vinho -> ovos, farinha	18.1489	82.5161
○	2	vinho -> ovos, farinha	9.9489	83.5161
○	3	agua, arroz, cola, fralda, ovos -> vinho	11.1489	89.1156
○	4	agua, arroz, fralda, ovos, farinha -> cola	17.1489	96.3235
○	5	agua, cola, fralda, vinho, ovos -> arroz	12.289	97.7612
○	6	arroz, cola, fralda, vinho, ovos -> agua	14.1489	100

Figure 1: PEAR screen showing some rules.

To help the user browsing a large set of rules and ultimately find the subset of interesting rules, we developed PEAR (Post processing Environment for Association Rules). PEAR implements the set of operators described below that transform one set of rules into another, and allows a number of visualization techniques. PEAR's server is run under an http server. One client is run on a web browser.

PEAR operates by loading a PMML representation of the rule set. This initial set is displayed as a web page (Figure 1). From this page the user can go to other pages containing ordered lists of rules with support and confidence.

To move from page (set of rules) to page, the user applies restrictions and operators. The restrictions can be done on the minimum confidence, minimum support, or on functions of the support and confidence of the itemsets in the rule. Operators can be selected from a list. If it is a $\{Rule\}\rightarrow\{Sets\ of\ Rules\}$ operator, the input rule must also be selected.

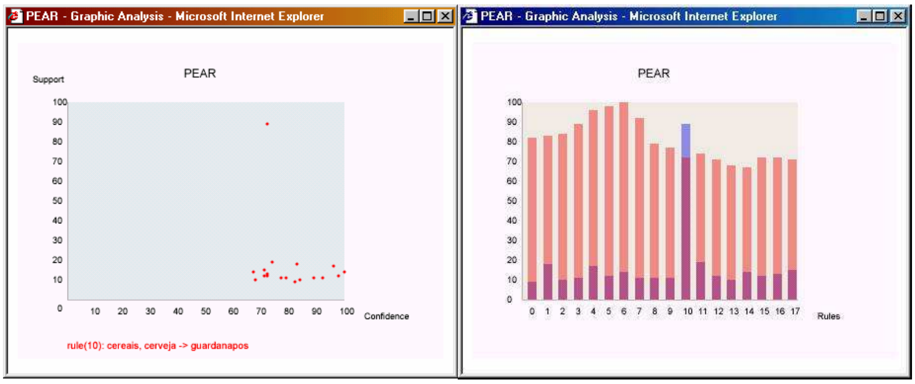

PEAR - Graphic Analysis - Microsoft Internet Explorer
PEAR
Support
100
90
80
70
60
50
40
30
20
10
0
10
20
30
40
50
60
70
80
90
100
Confidence
rule(10): cereais, cerveja -> guardanapos
PEAR - Graphic Analysis - Microsoft Internet Explorer
PEAR
100
90
80
70
60
50
40
30
20
10
0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
Rules

Figure 2: PEAR plotting support x confidence points for a subset of rules, and showing a multi-bar histogram.

For each page, the user can also select a graphical visualization that summarizes the set of rules on the page. Currently, the available visualizations are Confidence × Support plot and Confidence / support histograms (Figure 2).

4 Operators for sets of Association Rules

The association rule browser helps the user to navigate through the space of rules by viewing one set of rules at a time. Each set of rules corresponds to one page. From one given page the user moves to the following by applying a selected operator to all or some of the rules viewed on the current page. In this section we define the set of operators to apply to sets of association rules.

The operators we describe here transform one single rule $R\in\{Rules\}$ into a set of rules $RS\in\{Sets\ of\ Rules\}$and correspond to the currently implemented ones. Other interesting operators may transform one set of rules into another. In the following we describe the operators of the former class.

Antecedent generalization (AntG)

$AntG(A\rightarrow B) = \{A'\rightarrow B \mid A'$ is obtained by deleting one or more atoms in $A\}$

This operator produces rules similar to the given one but with a syntactically simpler antecedent. This allows the identification of relevant or irrelevant items in the current rule. The support and confidence lines of the resulting set of rules allow the visual identification of items to prune in the antecedent. In terms of the antecedent lattice, it gives all the rules below the current one with the same consequent.

Antecedent least general generalization (AntLGG)

$AntLGG(A \rightarrow B) = \{A' \rightarrow B \mid A'$ is obtained by deleting one atom in $A\}$

This operator is a stricter version of the *AntG.* It gives only the rules on the level of the antecedent lattice immediately below the current rule.

Consequent generalization (ConsG)

$ConsG(A \rightarrow B) = \{A \rightarrow B' \mid B'$ is obtained by deleting atoms in $B\}$

Consequent least general generalization (ConsLGG)

$ConsLGG(A \rightarrow B) = \{A \rightarrow B' \mid B'$ is obtained by deleting one atom in $B\}$

Similar to *AntG* and *AntLGG* respectively, but the simplification is done on the consequent instead of on the antecedent.

Antecedent specialization (AntS)

$AntS(A \rightarrow B) = \{A' \rightarrow B \mid A' \supseteq A\}$

This produces rules with lower support but higher confidence than the current one.

Antecedent least specific specialization (AntLSS)

$AntLSS(A \rightarrow B) = \{A' \rightarrow B \mid A'$ is obtained by adding one (any) atom to $A\}$

As *AntS*, but only for the immediate level above the current rule on the antecedent lattice.

Consequent specialization (ConsS)

$ConsS(A \rightarrow B) = \{A \rightarrow B' \mid B' \supseteq B\}$

Consequent least specific specialization (ConsLSS)

$ConsLSS(A \rightarrow B) = \{A \rightarrow B' \mid \mathrm{B'}$ is obtained by adding one (any) atom to $B\}$

Similar to *AntS* and *AntSS*, but on the consequent.

Focus on antecedent (FAnt)

$FAnt(A \rightarrow B) = \{A \rightarrow C \mid C$ is any$\}$

Gives all the rules with exactly the same antecedent. $FAnt(R) = AntG(R) \cup AntS(R)$.

Focus on consequent (FCons)

$FCons(A \rightarrow B) = \{C \rightarrow B \mid C$ is any$\}$

Gives all the rules with the same consequent. $FCons(R) = ConsG(R) \cup ConsS(R)$.

5 The Index Page

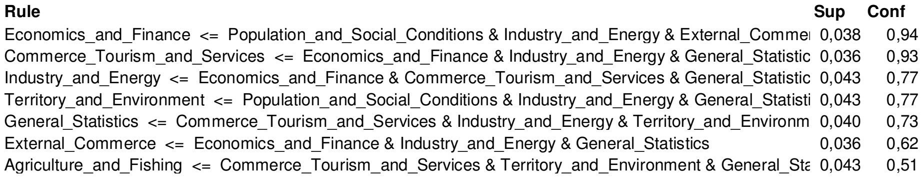
Our methodology is based on the philosophy of web browsing, page by page following hyperlinks. The operators implement the hyperlinks between two pages. To start browsing, the user needs an index page. This should include a subset of the rules that summarize the whole set. In terms of web browsing, it should be a small set of rules that allows getting to any page in a limited number of clicks. A candidate for such a set could be the, for example, the smallest rule for each consequent. Each of these rules would represent the lattice on the antecedents of the rules with the same consequent. Since the lattices intersect, we can change to a focus on the antecedent on any rule by applying an appropriate operator.

Similarly, we could start with the set of smallest rules for each antecedent. Alternatively, instead of the size, we could consider the support, confidence, or other measure. All these possibilities must be studied and some of them implemented in our system, which currently shows, as the initial page, the set of all rules.

6 One Example

We now describe how the method being proposed can be applied to browse through a set of association rules. The domain considered is the analysis of downloads done from the site of the Portuguese National Institute of Statistics (INE). This site (www.ine.pt/infoline) functions like an electronic store, where the products are tables in digital format with statistics about Portugal.

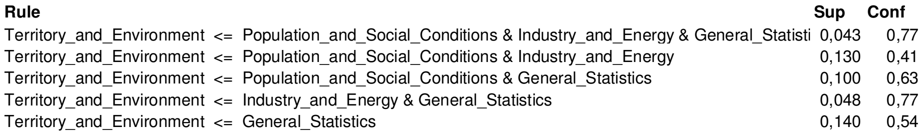
From the web access logs of the site's http server we produced a set of association rules relating the main thematic categories of the downloaded tables. This is a relatively small set set of rules (211) involving 9 items that serves as an illustrative example. The aims of INE are to improve the usability of the site by discovering which items are typically combined by the same user. The results obtained can be used in the restructuring of the site or in the inclusion of recommendation links on some pages. Although we show here how rules at the highest level of the products taxonomy, a similar study could be carried out for lower levels.



Rule	Sup	Conf
Economics_and_Finance <= Population_and_Social_Conditions & Industry_and_Energy & External_Comme	0,038	0,94
Commerce_Tourism_and_Services <= Economics_and_Finance & Industry_and_Energy & General_Statistic	0,036	0,93
Industry_and_Energy <= Economics_and_Finance & Commerce_Tourism_and_Services & General_Statistic	0,043	0,77
Territory_and_Environment <= Population_and_Social_Conditions & Industry_and_Energy & General_Statisti	0,043	0,77
General_Statistics <= Commerce_Tourism_and_Services & Industry_and_Energy & Territory_and_Environm	0,040	0,73
External_Commerce <= Economics_and_Finance & Industry_and_Energy & General_Statistics	0,036	0,62
Agriculture_and_Fishing <= Commerce_Tourism_and_Services & Territory_and_Environment & General_Sta	0,043	0,51

Figure 3: First page (index)

The rules in Figure 3 show the contents of one index page, with one rule for each consequent (from the 9 items, only 7 appear). The user then finds the rule on "Territory_an_Environment" relevant for structuring the categories on the site. By applying an operator, she can drill down the lattice around that rule, obtaining all the rules with a generalized antecedent.



Rule	Sup	Conf
Territory_and_Environment <= Population_and_Social_Conditions & Industry_and_Energy & General_Statisti	0,043	0,77
Territory_and_Environment <= Population_and_Social_Conditions & Industry_and_Energy	0,130	0,41
Territory_and_Environment <= Population_and_Social_Conditions & General_Statistics	0,100	0,63
Territory_and_Environment <= Industry_and_Energy & General_Statistics	0,048	0,77
Territory_and_Environment <= General_Statistics	0,140	0,54

Figure 4: Applying the operator ConsG (consequent generalization).

From here, we can see that "Population_and_Social_Conditions" is not relevantly associated to "Territory_and_Environment". The user can now look into rules with "Population_and_Social_Conditions" by applying the FAnt (focus on antecedent) operator. From there she could see what the main associations to this item are.

The process would then iterate, allowing the user to follow particular interesting threads in the rule space. Plots and bar charts summarize the rules in one particular page. The user can always return to an index page. The objective is to gain insight on the rule set (and on the data) by examining digestible chunks of rules. What is an interesting or uninteresting rule depends on the application and the knowledge of the user.

7 Implementation

To develop this web environment we chose a Microsoft platform, due to the development background of the team, and also because of the possibilities offered in terms of XML development. This option does not compromise our goal of having a browser-free tool. Currently, all PEAR's features are supported in both Netscape and Internet Explorer. In the following sections we describe the main technologies involved in PEAR. The interactions are summarized in Figure 5.

Microsoft Internet Information Server

We use the Microsoft Internet Information Server (IIS) as PEAR's http server to run the Active Server Pages (ASP) for server-side programming, allowing database and XML manipulation and form data submitted by the user. PEAR also runs offline with no limitation under Microsoft Personal WebServer (Windows 95/98/2000/Me) or under Microsoft Peer Web Services (Windows NT Workstation).

Active Server Pages, VbScript and JavaScript

Active Server Pages (ASP) (Microsoft) are dynamic and interactive web pages processed on the server-side, thus useful to manipulate data submitted by users (for instance, selecting a set of association rules given certain restrictions by the users) as well as manipulating database requests. An ASP page integrates HTML tags with script commands. These scripts can be either VbScript or Jscript (JavaScript similar).

PEAR uses VbScript (in Active Server Pages) to process the set of association rules represented in a PMML document (using Document Object Model), to allow the user to browse through it, and to store the rules in a relational database. JavaScript, (Microsoft Web Site) is used for data manipulation on the client-side, for its portability. With JavaScript we create and manipulate PMML documents or SVG (both XML documents) using Document Object Model.

The Document Object Model (DOM) is a tree structure-based application program interface (API) for HTML and XML documents issued as a W3C Recommendation in October 1998 (W3C DOM Level 1 specification). It is used to process and manipulate an XML document by accessing its internal structure. DOM represents an XML document as a tree. Its nodes are elements, text, and so on. DOM makes it convenient for application programs to traverse the tree and access the contents of the tree.

We use a relational database (Microsoft Access) to store the PMML model and take advantage of using Structured Query Language (SQL) to obtain sets of association rules.

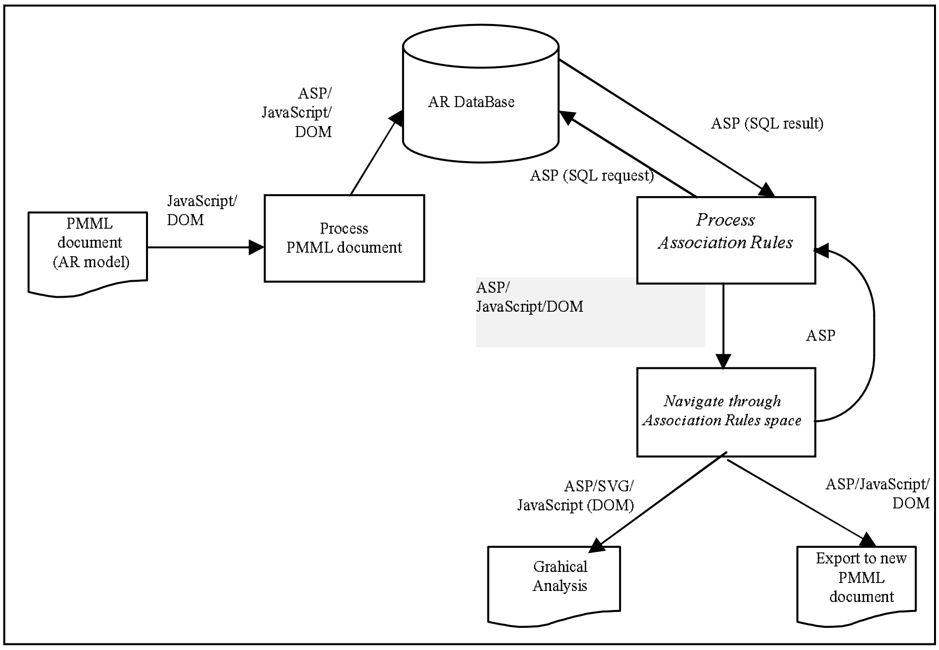


Figure 5: General architecture of PEAR.

Scalable Vector Graphics and PMML

Scalable Vector Graphics (SVG) is an XML-based language that specifies and defines vector graphics that can be visualized by a web browser. [W3C Recommendation]. We manipulate SVG graphics with VbScript or JavaScript (using Document Object Model). With SVG, it is easy to produce a data visualization and even make it interactive (controlling keyboard or mouse events). PEAR gets data from PMML and presents it using VbScript and SVG graphics.

Predictive Model Markup Language (PMML) is an XML-based language. A PMML document provides a non-procedural definition of fully trained data mining models with sufficient information for an application to deploy them. It provides a way for people to share models between different applications. Like any XML document, also a PMML document must follow a Data Type Definition (DTD) that defines the entities and attributes for documenting a specific data mining model. A model described using PMML has the following structure:

1) A header,
2) A data schema,
3) A data mining schema,
4) A predictive model schema,
5) Definitions for predictive models,
6) Definitions for ensembles of models,
7) Rules for selecting and combining models and ensembles of models,
8) Rules for exception handling.

The main reasons that drove the formulation of the PMML for predictive models were that it must be extensible, portable and human readable. It allows users to develop models within one vendor's application, and use other vendors' applications to visualize, evaluate or otherwise use the models. PEAR can read an AR model specified in PMML. The user will be able to manipulate the model, creating a new rule space based on a set of operators, and export a subset of selected rules in PMML.

8 Related Work

There is some work on the visualization and summarization of association rules. In this section we refer to selected work on theme.

The system DS-WEB (Ma et al.) uses the same sort of approach as the one we propose here. In common, DS-WEB and PEAR have the aim of post processing a large set of AR through web browsing and visualization. DS-WEB relies on the presentation of a reduced set of rules, called direction setting or DS rules, and then the user can explore the variations of each one of these DS rules. In our approach, however, we rely on a set of general operators that can be applied to any rule, including DS rules as defined for DS-WEB. The set of operators we define is based on simple mathematical properties of the itemsets and have a clear and intuitive semantics. PEAR also has the additional possibility of reading AR models as PMML.

VizWiz is the non-official name for a PMML interactive model visualizer implemented in Java (Wettshereck). It graphically displays, not only association rules, but many other data mining models. The philosophy of WizWiz for displaying AR relies on the presentation of the list of rules, allowing the user to set the minimal support and confidence through very intuitive gauges. VizWiz also accompanies the display of each rule by color bars representing support and confidence. This visualizer can be used directly in a web browser as a java plug-in.

(Lent et al 97) describe an approach to the clustering of association rules. The aim is to derive information about the grouping of rules obtained from clustering. As a consequence one can replace clustered rules by one more general rule.

9 Future Work and Conclusions

Association rule engines are often rightly accused of overloading the user with very large sets of rules. This applies to any software package, commercial or non-commercial, that we know.

In this paper we describe a rule post processing environment that allows the user to browse the rule space, organized by generality, by viewing one relevant set of rules at a time. A set of simple operators allows the user to move from one set of rules to another. Each set of rules is presented in a page and can be graphically summarized. In the following we summarize the main advantages, limitations and future work of the proposed approach.

As main advantages of our approach, we can say that PEAR enables selection and browsing across the set of derived AR, as well as plotting numeric properties of each subset of rules found. Browsing is done by a set of well-defined operators with a clear and intuitive semantics. This way, the selection of AR rules by an user is an implicit form of providing background knowledge, that can be later used, for example, in selecting rules for a a classifier made out of a subset of rules. PEAR presents an open philosophy by reading the set of rules as a PMML model.

The main limitations are that visualization techniques are always difficult to evaluate. This one is no exception. Moreover, the current implementation requires, on the server-side, the use of an operating system from one specific vendor.

For future work, we intend to develop metrics to measure the gains of this approach, as well as mechanisms that allow the incorporation of user defined visualizations and rule selection criteria, such as for example, the combination of primitive operators, and implement other visual representations of subsets of rules. It is also important to evaluate the current implementation against other alternatives such as java, as well as an alternative architecture to client-server, such as plug-in.

References

Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., Verkamo, A. I., Fast Discovery of Association Rules. *Advances in Knowledge Discovery and Data Mining*: 307-328. 1996.

Data Mining Group (PMML development), http://www.dmg.org/

Lent, B., Swami, A., Widom, J.: Clustering Association Rules, in Alex Gray, Per-Åke Larson (Eds.): Proceedings of the Thirteenth International Conference on Data Engineering, ICDE 97 Birmingham U.K. IEEE Computer Society 1997

Ma, Yiming, Liu, Bing, Wong, Kian (2000), Web for Data Mininig: Organizing and Interpreting the Discovered Rules Using the Web, School of Computing National University of Singapore, SIGKDD Explorations, ACM SIGKDD, Volume 2, Issue 1, July 2000.

Microsoft Web Site (Active Server Pages) http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnbegvb/html/activeserverpages.asp

Microsoft Web Site (Descriptions of Java, JScript, and JavaScript) http://support.microsoft.com/default.aspx?scid=kb;EN-US;q154585

W3C, Scalable Vector Graphics (SVG) 1.0 Specification, W3C Recommendation September 2001, http://www.w3.org/TR/SVG/

W3C DOM Level 1 specification http://www.w3.org/DOM/

Wettshereck, D., Sol-Eu-Net deliverable, http://soleunet.ijs.si, 2001.