A Software Architecture for Data Pre-Processing using Data Mining and Decision Support Models

Marko Bohanec[1], Steve Moyle[2], Dietrich Wettschereck[3], Petr Mikšovský[4]

[1] Jožef Stefan Institute, Jamova 39, SI-1000 Ljubljana, Slovenia
marko.bohanec@ijs.si
[2] Oxford University Computing Laboratory, Wolfson Building, Parks Road, Oxford, OX1 3QD, United Kingdom.
steve.moyle@comlab.ox.ac.uk
[3] Fachhochschule Bonn-Rhein-Sieg, Grantham-Allee 20, 53575 Sankt Augustin, Germany
dietrich.wettschereck@fh-bonn-rhein-sieg.de
[4] Czech Technical University, Technická 2, Prague 6, CZ 166 27, Czech Republic
miksovsp@labe.felk.cvut.cz

Abstract. A new software architecture is proposed that integrates several key data mining and decision support techniques used in the SolEuNet Project. The software is aimed at providing data-pre-processing services to Zeno-for-RAMSYS, a methodology and support system for use in rapid remote collaborative data mining projects. The architecture extends Sumatra, a scripting language for the specification of data transformation tasks, with the capability to use various data mining and decision support models, for data pre-processing - for instance decision trees and hierarchical multi-attribute models. The integration of models takes place at the level of representation, which is common for all models and based on PMML, an emerging standard for sharing models.

1 Introduction

The motivation and key premise to this paper is the proposition that to solve any data analytic problem requires the appropriate level of abstraction from the real world of the problem, to the data structures that represent the problem for the analysis. Many researchers (e.g. [9]) claim that the ease of analytic problem solving depends on achieving the appropriate level of abstraction for the problem. Often the data available to solve a problem is not in a form (or abstraction) that is appropriate. To resolve this difficulty it is necessary to either gather extra data, transform the original data, or transform both the original and any extra data. It is often considered that having obtaining the "correct" transformation of the data, the problem is almost solved. This paper is about data transformations also called data pre-processing with respect to two data driven problem solving techniques: Data Mining and Decision Support.

The SolEuNet Project[1] aims to promote and integrate the two IT fields of Data Mining and Decision Support within Europe. The Project Consortium contains experts from the two fields spread across fourteen partner institutions from seven countries. To achieve its aims the SolEuNet Project is using the combined expertise of its partners to solve real-world problems from industry. Currently, the Project is actively attracting and solving real world problems using expertise from data mining and decision support.

One of the developments of the SolEuNet Project is a data pre-processing tool called Sumatra TT [1]. Sumatra TT (Transformation Tool) is a metadata-driven, platform independent, extensible, and universal data processing tool. These features have been achieved by building the tool as an interpreter of the transformation-oriented scripting language called Sumatra. The Sumatra language is a fully interpreted Java-like language combining together data access, metadata access, and common programming constructions. Furthermore, it supports RAD (Rapid Application Development) technology via a library of re-usable transformation templates.

To meet its aims the SolEuNet Project must tackle many of the following problems: remote collaborative problem solving; sharing, evaluating and combining multiple solutions to data analytic problems; combining the techniques and expertise of data mining and decision support experts; and many mundane non-technical tasks (e.g. the provision of legal and contractual arrangements with the industrial problem owners).

This paper proposes a novel architecture that is fundamental to the success of the SolEuNet Project. It is centered on the ability for data analytic experts (here the experts are drawn from the areas of data mining and decision support) to be able to communicate and share their efforts by being able to describe, share, and execute data pre-processing tasks. The approach is based upon the common use of PMML models to describe three types of models: 1) data mining models, 2) decision support models, and 3) data transformation models.

The paper is structured as follows. The key techniques and building blocks for the SolEuNet Project are described in Section 2. Here the focus is on methods for describing and sharing data analytic models. Section 4 proposes a way of using such models for data preprocessing, while Section 3 suggests a software architecture for achieving such functionality. Section 5 concludes the paper.

2 Key Techniques and Building Blocks

This section introduces five key techniques for the data pre-processing architecture. These include a language for describing and executing data transformations, models used in decision support, models generated by data mining, a standard model description language, and a methodology for remote collaborative data mining called RAMSYS, and a system to support it.

[1] The IST-1999-11495 project Solomon European Network - Data Mining and Decision Support for Business Competitiveness: A European Virtual Enterprise.

2.1 Data Pre-processing

Often, there is a data mining or decision support problem with supporting data, which may have been collected without considering further analysis. The first necessary step before any analytical algorithm can be applied is the transformation of such data into an appropriate form. This process is usually called data pre-processing [12] in data mining systems and data transformation in decision support systems. Although the name of the process and target areas is different, the problems to be solved in this context are similar (e.g. transporting data among different formats and platforms, calculating statistical characterization of the data, grouping data sets, splitting data sets, etc.).

The Sumatra [1] language has been designed as a special, fully interpreted, metadata-driven, transformation-oriented scripting language. Sumatra emphasizes the following: simple usage from the programmer's point of view, simple parsing and easy integration into a data transformation system. The language syntax is inspired by those of C++ and Java. It supports all commonly known programming structures, and in particular it promotes code reuse. Sumatra has a built-in set of objects and functions most of which ensure data access, metadata access, and basic reporting features. Both, data access objects and metadata access objects define a standardized interface for extensibility.

The language itself is platform and data source independent. Due to the fact that metadata access is integrated into the scripts, the user can design transformations regardless of the types of data sources. Furthermore, the Sumatra language supports RAD (Rapid Application Development) technology by providing a library of re-usable transformation templates. Templates are, in fact, skeletons of Sumatra scripts. The Sumatra language is interpreted by Sumatra TT (Transformation Tool). This tool ensures that all the language features can be easily exploited for real-world problems.

Every pre-processing task realized using Sumatra TT consists of design and run-time phases. The design phase means the definition of all data sources and the development of transformation scripts on the client side. A typical user who is an expert in data mining or data warehousing but who is not a programmer can take advantage of the graphical user interface (GUI). The GUI interactively allows both data definition and script development. The run-time phase corresponds to the execution of the script on the server side. From the user's perspective, the execution can be invoked immediately or scheduled for later running.

The system has been successfully tested and used in several real life applications for both data transformation and data mining tasks. It allows a quick and simple preparation of data transformations thanks to a wide range of formatting capabilities as well as the power of the Sumatra language and its templates and macros. The well defined structure of data interfaces allows the development of further tools and extensions.

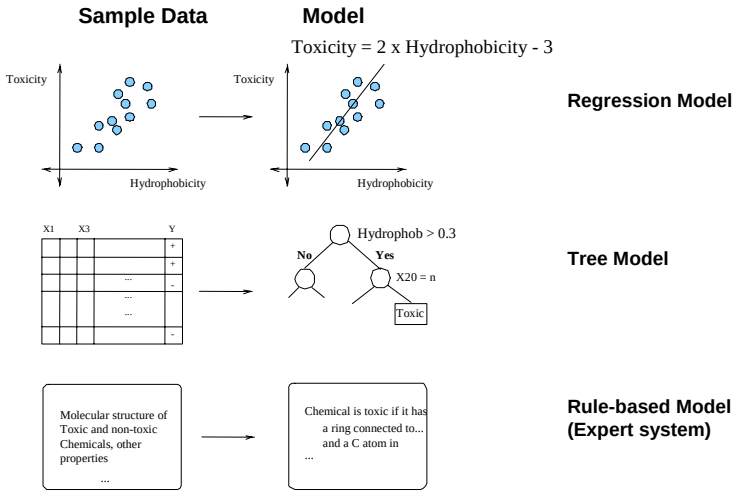

Fig. 1. The structure of three types of models produced by data mining.

2.2 Data Mining Models

Data mining is about using sample data to produce models. The types of models that are produced depend on the style of data mining task - e.g. classification, prediction, description, subgroup-discovery, associations, regression, clustering – as well as the type of algorithm used. There are many and varied data mining model types for instance: - regression models, decision trees, clusters, association rules, Horn clauses, and neural nets. Three types of models produced from sample data are illustrated in Fig. 1.

When the data mining task is to forecast or predict the value of some attribute(s) of future, unseen instances, the model provides a functional mapping from input values to the attribute(s) value(s). The utility of the data mining models produced is often measured in the accuracy of the predictions. Some researchers [11] also regard the ability to understand the model as being particularly valuable. For a model to be understood it must be able to be communicated to a human in some way. Different data mining model types have different levels of human intelligibility. The definition of intelligibility varies from problem to problem, as well as from person to person and is a subjective area. However, when working collaboratively, it is important to be able to understand the results and efforts of other people. Models that are able to be easily communicated and understood are to be preferred to those that cannot (provided the appropriate utility criteria are also met).

The example data mining problem shown in Fig. 1 is that of producing models for the prediction of toxic chemicals. The first model type is that of linear regression in which the amount of toxicity of any particular chemical is predicted based on the measurement of that chemical's hydrophobicity. In the

second decision-tree model, the categorical prediction of a chemical's toxic nature is determined by testing the values of each attribute mentioned in the internal nodes of the tree. The prediction itself is contained in the leaf node. A tree structure orders the attributes in some form of importance, which in itself can lead to an understanding of the model's features. The final model is that of a collection of rules, which are capable of being presented in natural language for easier comprehensibility. To achieve a high level of comprehensibility, however, the models may further need to be transformed into a form that is natural for the problem owner. For example, it may be necessary to use pictorial representations of the instances that highlight the relationships contained within the model [7].

2.3 Decision Models

Decision models originate in operations research and decision analysis. Operations research [10] is concerned with optimal decision making in, and modelling of, deterministic and probabilistic systems that originate from real life. Typical techniques include linear and nonlinear programming, network optimization models, combinatorial optimization, multi-objective decision making, and Markov analysis. Decision analysis [5] provides a framework for analyzing decision problems and usually proceeds by building models and using them for various analyses and simulations, such as "what-if" and sensitivity analysis, and Monte Carlo simulation. Typical decision analysis modelling techniques include decision trees, influence diagrams, and multi-attribute utility models.

Let us illustrate the approach by a real-world hierarchical multi-attribute model for risk assessment in diabetic foot care [3]. Based on the results of screening of diabetic patients, the model attempts to assess the risk of developing their foot pathology by classifying them into four risk groups. This assessment is an important indicator for prescribing a patient's therapy, organizing further screening and educational activities, and monitoring the development of the disease. The model has been developed in collaboration between a medical doctor and two decision analysts using DEX, a tool for developing qualitative hierarchical models [2]. The structure of the model is shown in Fig. 2. There are three levels of attributes. The seven basic ones correspond to seven measurements that are taken during the patient's screening. Ulcers, for example, describes the placement and severity of already developed ulcers on the patient's foot. The topmost attribute RISK represents the four risk groups. The three intermediate aggregate variables correspond to three risk assessment subproblems: History of disease, Present status of the patient's foot, and the results of screening Tests.

In decision practice, such a model is "normally" used so that a patient is examined, and data corresponding to the seven basic attributes of the model are obtained. These represent the input variables to the evaluation of risk. Their values are then aggregated yielding an estimated level of risk, represented by one of the four levels of the root attribute. The evaluation procedure also assigns values to the three intermediate variables History, Present status, and Tests.

In Section 3 we show how such a model can be used in an "unusual" way so as to contribute to data pre-processing.

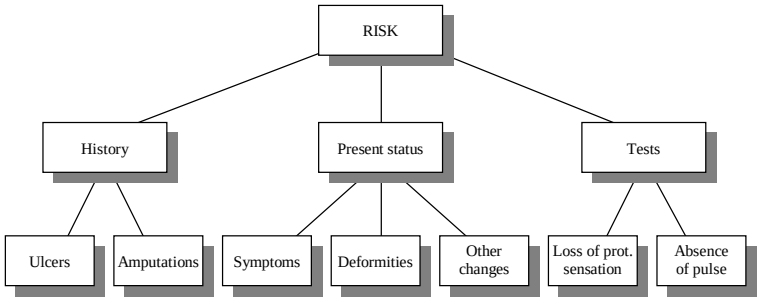

Fig. 2. The structure of a model for risk assessment in diabetic foot care.

2.4 PMML

The Predictive Model Markup Language (PMML) is a set of Document Type Descriptions (DTDs) specified in XML. The first version (1.0) was provided in July 1999 [6, 8] by the Data Mining Group (DMG). The DMG is "an independent, vendor-led group which develops data mining standards". Currently, its seven core members are Angoss, IBM, Magnify, NCR, Oracle, SPSS, and the National Centre of Data Mining (University of Illinois, Chicago). Additionally, a number of associated members have been accepted to the group. Version 1.1 has been released recently to incorporate improvements based on the lessons learned from the first release as well as definitions for further data mining models. Version 1.2 is currently under development.

The advantages of PMML can be summarized as follows (see also paper by Wettschereck & Müller in these proceedings (if accepted)).

- It provides independence of the knowledge extracted from application, implementation, (hardware) platform, and operating system.
- It simplifies the use of data mining models by other applications or people. For example, consultants or researchers (SolEuNet [13] members etc.) can function as producers of models and customers can import models into their own tools.
- It is not concerned with the process of creating a model or the specific implementation of the algorithm.
- DTDs support proprietary extensions to allow for enriched information storage for specialized tools.

Fig. 3 shows the quite simplified data mining process. The PMML models are typically produced by data mining algorithms. For implementations that do not directly support PMML, converters can be employed to translate the proprietary format into PMML. The advantage of using the open standard PMML format is that models can subsequently be used by other applications or data mining tools. In this way, data mining models are portable across platforms and applications.

The PMML 1.1 definition includes the following types of models:

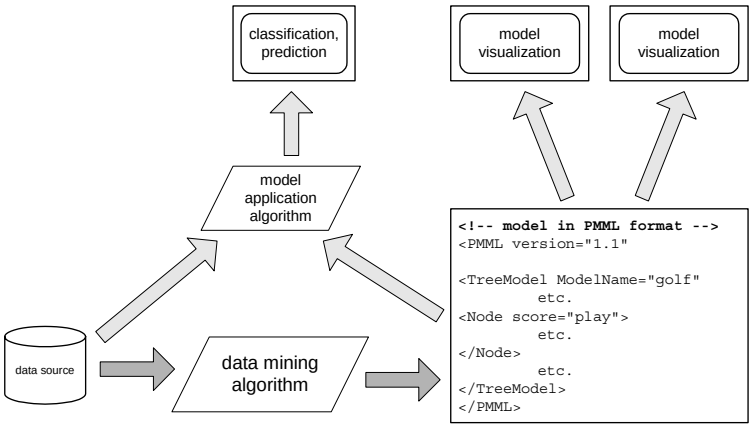

Fig. 3. The role of PMML in the KDD process. After a model is generated by a data mining algorithm and stored in the PMML format, it can be used by other tools to visualize the model or to classify unseen values. Thanks to this open standard format, tools from different vendors can be used by different users or at different stages of the data mining process.

– polynomial regression – general regression – decision trees – center based clusters – density based clusters – associations – neural nets

In principle, it is possible to specify PMML DTDs for all possible data mining models and even for all conceivable data pre-processing steps. The DMG will continue its work on producing new DTDs for additional model types and on extending existing DTDs to accommodate for a wider range of models.

2.5 Zeno-for-RAMSYS

RAMSYS is a methodology that enhances the standard data mining methodology CRISP-DM [4] so that data mining may be performed by a team of experts collaborating remotely. Zeno-for-RAMSYS (explained below) is a support system for the methodology. The key principles of the RAMSYS methodology to enable Remote Collaborative Data Mining are the following.

Principle 1: Share all Knowledge This is the foundation of the RAMSYS methodology. At the start of a data mining problem all the assembled information should be made accessible to the data mining experts. For any data mining problem one of the main sources of information is a database (or data set).

Principle 2: Stop Anytime This is *key* to delivering a solution to a data mining customer. This states that at any point in the problem solving process the solutions (or partial solutions) generated can be assembled and assessed.

Here there is a necessity to define strategies for combining and assessing the alternative solutions produced from within the data mining team.

Principle 3: Start anytime This will have obvious value, particularly when extra data mining expertise is drafted into the problem solving team, some time after the process is started. This, however, can only be achieved if the knowledge about the problem is shared in a way that makes the current best understanding available.

Principle 4: Problem Solving Freedom This becomes paramount as there are a multitude of data mining techniques available - the data mining experts should apply those techniques that they have most expertise in using and/or are most appropriate to the problem itself. Often in a data mining problem there are subtasks that can be approached pseudo-independently this then will benefit from the next principle.

Principle 5: Task Assignment This provides some managerial type of control over the problem solving process. Often in a data mining problem there are subtasks that can be approached pseudo-independently. For example, data pre-processing is a key task in a data mining project. Some experts are more skilled at this task compared to that of working with the data to produce solutions (or "models").

The RAMSYS methodology is supported by the group-ware system Zeno [15]. The Zeno II system is tailored towards remote collaborative data mining. The development of Zeno-for-RAMSYS will provide the following key functionality: *collaboration*, *communication*, and *awareness*. The system will contain the following types of information: documents relating to the problem situation; meta data relating to the dataset; descriptions of lines of enquiry being pursued by the data mining experts; descriptions of transformations of the data set and their efficacy; partial/complete results. In particular, the transformations and the results deserve elaboration.

The transformation of the data set are typically motivated by the need to convert the data set into a form that matches the data input requirements of different algorithms (or "modelling techniques"). As mentioned above, transformations are a method of altering the abstraction of the problem. Such transformations (performed by data pre-processing) are often functional transformations from one view of the dataset into another. For example the common pre-processing operation of discretisation takes an attribute on the real domain, and transforms it to a new attribute on the integer domain. Such transformations need to be shared within a collaborative data mining project. Early evidence of collaborative data mining (see also paper by Voß, Gärnter & Moyle in these proceedings (if accepted)) demonstrates that experts in pre-processing have produced modified datasets that were subsequently used by many modelling experts.

3 Using Models for Data Transformations

There are many types of data mining (see Section 2.2) and decision models (Section 2.3) that define a mapping from a set of input variables to one or

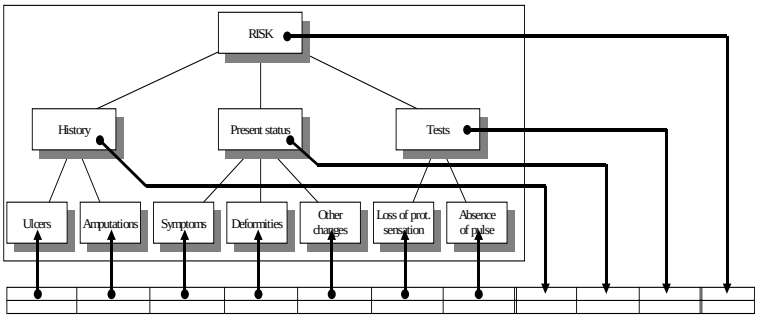

Fig. 4. The diabetes model used as a data transformation mechanism.

more output variables. Usually, such models are used by experts or embedded into systems to perform various classification, evaluation, and analysis tasks. However, their mapping properties can also be used in data pre-processing for performing data transformations. In principle, any model can transform data items corresponding to its input variables to one or more aggregate values that correspond to its outputs. Good models are expected to generate new data items that would be useful for further data analysis.

For example, take the diabetes model from Section 2.3 and recall that it maps seven input variables into three intermediate and one final, which represents the level of risk for developing foot pathology. Suppose there is a data set containing patients' data, including the seven measurements required for risk assessment. Then, the model can be used to generate up to four new database fields corresponding to its output variables, as shown in Fig. 4. These transformations aggregate data and follow the rules defined in the model by the expert, so they in a way encode his background medical knowledge. Consequently, the new data items are expected to be important and contribute considerably to the quality of data mining using the extended data set.

4 Software Architecture

In order to facilitate the application of models for data transformation, we propose software architecture that integrates all the key techniques presented in Section 2. The central point of this architecture (Fig. 5) is Zeno-for-RAMSYS (see Section 2.5) as a system for remote collaborative data mining. In order to perform its data transformation tasks, Zeno-for-RAMSYS will use the Sumatra language and pre-processing engine (Section 2.1). Finally, to allow the application of Data Mining (Section 2.2) and Decision Support Models (Section 2.3), Sumatra will have to be extended by the capability to import and export external models and use them for data transformation. The external models can be developed using some other software products, either as a result of data analy-

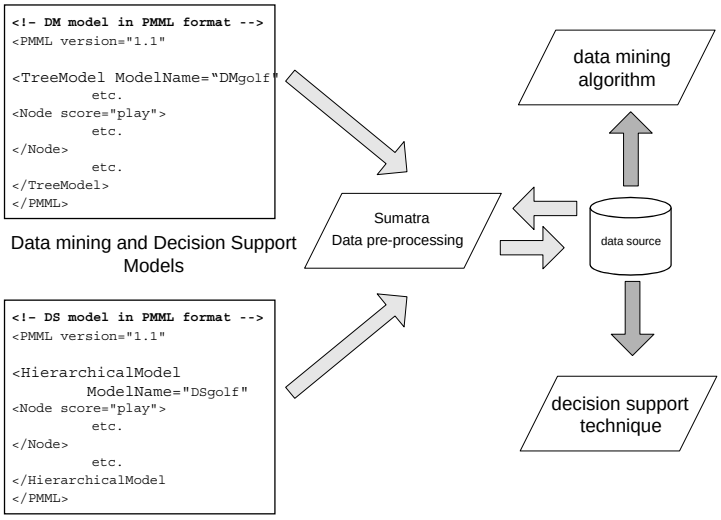

Fig. 5. A software architecture for data transformation using Data Mining and Decision Support models.

sis, or based on expert modelling. To facilitate an easy interchange of models, models should be represented in PMML (Section 2.4).

Representing data transformations in PMML is a natural extension of both Sumatra and the PMML 1.1 standard. PMML is based on the extensible Markup Language (XML) [14]. At this point in time the Sumatra interactive tool SumatraTT, which provides a graphical user interface that enables the creation of Sumatra scripts, stores much of its information in XML data structures. Furthermore, there are already "model" types defined in PMML that describe forms of transformations.

The normalization PMML subset provides a document type definition (DTD) for a particular form of data transformation. DTD syntax is part of the XML 1.1 standard [14]. The PMML 1.1 documentation [6] states the "DTD subset for normalization provides a basic framework for mapping input values to specific value ranges, usually the numeric range [0 .. 1]. It is used by the DTD for neural networks. Similar instances are also used in regression models." This is clearly a transformation from input values to output values.

We now consider the potential advantages of such an architecture, particularly with respect to collaborative problem solving, and data mining and decision support. In addition to a common style of model representation (i.e. PMML) we also have the executable Sumatra scripts. Furthermore the system provides a number of other tools and formalisms that can be used for the development of transformations. It provides a flexible, consistent, and principled development and interchange of models. The system allows the extension of data sets offer-

ing the potential of improved quality of subsequent data mining and decision support processes.

The architecture provides one approach to combining data mining and decision support. It would enable the "partial solution" models from each to be used as input to the others' problem solving processes. That is data mining models from partial solutions of the problem can be utilized as input to the decision support process, and vice versa.

The ability of the SumatraTT to process PMML models would make evaluation of each potential solution (or partial solution) easier. For example, when trying to assess a number of competing problem solutions, each being a model described in a PMML format, the SumatraTT tool could apply each of these models to the previously unseen evaluation data which was initially set aside, and could then produce comparison statistics (e.g. confusion matrices) for each model.

Furthermore, the architecture could be employed as one potential model combination strategy. It would be possible to use Sumatra scripts to integrate many alternative models. Here SumatraTT descriptions could be used to describe and define the manner in which several PMML models could be combined - perhaps guided by some decision surface like ROC.

Such potential is not without its difficulties. First, there is only limited definitions for model types supported by the PMML standard. In particular, there are no decision support model types defined by the PMML standard. Significant effort would be required to provide such definitions and have them accepted as part of any standardization process. There are also many software extensions that will need to be developed. One such is that the Sumatra system would need to be extended to cope with models expressed in the PMML format. Second, the Zeno-for-RAMSYS collaboration system will need to have explicit support for PMML models and the Sumatra tools.

5 Conclusion

When working collaboratively on any data analytic problems solving it is important to be able to share results in a uniform manner. In any form of data analytic problem it is essential to be able to transform the data into alternative forms to suit both the problem and the techniques for solving the problem. Data mining models can be represented using PMML (an extension of XML). It was suggested that decision support models could also be represented by PMML formats (possibly requiring extensions to the standard version). Both data mining and decision support types of models can, sometimes, describe functional mappings from input values to output values. Similarly, data pre- processing performs mappings form input values to output values. It was suggested that maybe data pre-processing could utilize data mining and decision support models.

The data pre-processing language called Sumatra was described. The Sumatra interpreter (executing tool), Sumatra TT has been introduced. Its first release is already capable of utilizing XML format representations. It was proposed that

the SumatraTT tool could be extended to utilize some PMML models. This is quite a natural extension of the PMML concept. Indeed, some data transformations are already describable in PMML 1.1 (e.g. normalization).

One application of such an architecture with the Zeno-for-RAMSYS system would be the to utilize it to evaluate alternative PMML models submitted by different data mining and decision support experts in a consistent manner.

6 Acknowledgment

The work reported here was supported in part by the EU project SolEuNet, IST- 11495, and the Slovenian Ministry of Education, Science and Sport.

References

1. Aubrecht, P., Kouba, Z.: *Metadata Driven Data Transformation.* to be published in the proceedings of SCI 2001. Orlando (Florida) : International Institute of Informatics and Systemics (2001).
2. Bohanec, M., Rajkovič, V.: *DEX: An expert system shell for decision support.* Sistemica 1(**1**) pp 145 - 157 (1990).
3. Bohanec, M., Zupan, B., Rajkovič, V.: *Applications of qualitative multi-attribute decision models in health care.* International Journal of Medical Informatics 58 - 59 pp 191 – 205 (2000).
4. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R.: *CRISP-DM 1.0: Step-by-step data mining guide.* CRISP-DM consortium, (2000).
5. Clemen, R.T.: *Making Hard Decisions: An Introduction to Decision Analysis.* Duxbury Press (1996).
6. http://www.dmg.org
7. Finn, P., Muggleton, S.,Page, C.D., and Srinivasan, A.: Pharmacophore discovery using the inductive logic programming system Progol. *Machine Learning*, 30:241-271, (1998).
8. Grossman R. L., Bailey, S., Ramu A., Malhi, B., Hallstrom, P., Pulleyn, I., Qin, X.: *The Management and Mining of Multiple Predictive Models Using the Predictive Modeling Markup Language (PMML)*, Information and Software Technology, Volume 41, pages 589-595 (1999).
9. Han, J., and Kamber, M.: *Data Mining: Concepts and Techniques.* Morgan Kaufman, (2001).
10. Hillier,F.S., Lieberman, G.J.: *Introduction to Operation Research.* McGraw Hill (2000).
11. Michie, D.: *Machine Learning in the Next Five Years.* In D. Sleeman and J. Richmond editors, Third European Workshop Session on Learning (EWSL '88), p. 107 - 122. University of Strathclyde. Pitman (1988).
12. Mikšovský, P., Štěpánková, O.: *Data Pre-processing for Data Mining.* Research report, Czech Technical University, Gerstner Lab., GL 108/00, 8 p. (2000).
13. SolEUNet: Data Mining and Decision Support for Business Competitiveness: A European Virtual Enterprise. http://soleunet.ijs.si/website/html/euproject.html
14. Extensible Markup Language (XML) 1.0 (Second Edition). http://www.w3.org/TR/2000/REC-xml-20001006
15. Zeno consensus building. http://zeno.gmd.de/MS/index.html