Improved Architectural Support for Analysis and Design of Data Models

Plamena Ts. Andreeva[1], Svetla I. Vassileva[2] and Valentin S. Stoyanov[3]

Abstract **– In decision – making process new architectural support is needed for the design of data models. This paper aims to present the advantages of visual architectural support covering the whole software development life cycle. We propose new methodology for integrating data flow diagrams and techniques for data analysis. The experimental methodology is applied to the previously explored data models.**

Keywords **– Data analysis, Model design, Support Vector Machines, Architectural support.**

I. Introduction

As computer technology is increasingly being applied in real measurement and control tasks, new methods have been developed [2] for medical diagnosis, education, and training. In order to assist decision – making process new architectural support is needed for the design of data models. Such architectural support provides a technological context for understanding the information design process.

The machine learning (ML) technology provides systems with new tools able to improve process supervision. One useful tool for processing information and analyzing feature relationships is data mining (DM) technique [1]. In the field of artificial intelligence and ML such modern technologies [8] have the potential to put decision – making and prediction planning on a more quantitative footing through the combination of extraoperative simulation and intraoperative image guidance.

The conducted experiments have in common a body of knowledge concerning experimental methodology that specifies how to design `good' data model. To avoid dubious comparison between algorithms and lack of suitable quantification of performance and its variability an improved architectural support is needed. In this paper we address the visual flow diagram towards the model driven approach. This methodology provides support to requirement capturing, analysis and design. Changes in model or source code can always be synchronized.

II. Visual representation in Knowledge analysis

The automated acquisition of knowledge by ML approach is an active area of current research [4], [6]. The fundamental problem still comes down to a human interface issue. The methods for data analysis embedded in the black-box software currently available makes their misuse proportionally more dangerous. Different approaches [7], for example, or models may be derived that are built upon wholly specious assumptions. Therefore, an understanding of the statistical and mathematical model structures underlying the software through visual support [12] is promising improvement.

In simple scenarios, there often appears at first to be little difference between the way entities are represented in the application (as objects) and in the database (as rows in tables). When new tables are created, the state of data objects in the table is stored. An example of Unified Modeling Language (UML) class diagram is shown in fig.1 and the correspondding entity-relationship diagram for the database is produced.

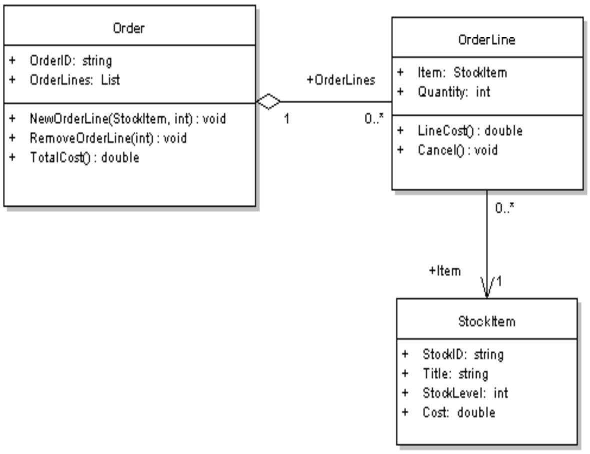

Figure 1. UML class diagram for a simple scenario.

To assist the overall decision-making process a visual data flow is presented with the modular architecture (as shown on fig. 3). This allows separation of core functionality and creation of separate models for specific data analyses. In a typical object oriented environment, each object is automatically assigned a unique ID by the system. The value of this internal ID does not depend on the values of any of the fields. Defining relationships between objects gives the structure for integrating data diagrams and data models to which they are related.

[1] Plamena Ts. Andreeva is with the Department of Knowledge Based Systems at the Institute of Control and System Research, Bulgarian Academy of Sciences, 1113 Sofia, Bulgaria, E-mail: plamena@icsr.bas.bg

[2] Svetla I. Vassileva is the head of the Department of Knowledge Based Systems at the Institute of Control and System Research, Bulgarian Academy of Sciences, 1113 Sofia, Bulgaria, E-mail: vasileva@icsr.bas.bg

[3] Valentin S. Stoyanov is with the Department of Knowledge Based Systems at the Institute of Control and System Research, Bulgarian Academy of Sciences, 1113 Sofia, Bulgaria, E-mail: v_sto@icsr.bas.bg

III. The Domain Model

The sample application has a very simple domain model based on the state of the system. The classes contain properties and methods. A model developer creates the models using class diagram or entity relationship diagram and generates the executable persistence layer from the models. With the sophisticated model-code generator, the persistent model will be updated automatically according to any modification.

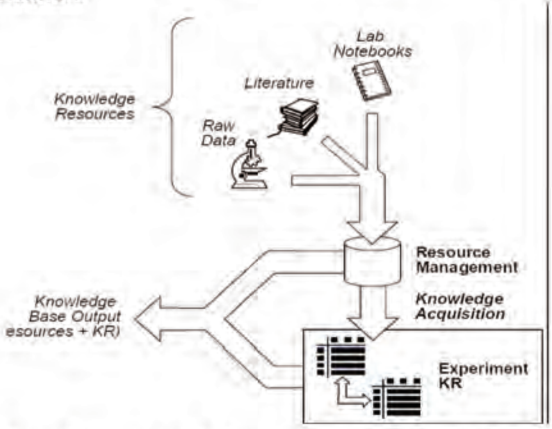

Figure 2. The overall scheme of the domain model design

To make objects persistent, their state values must be saved. The fields of each class are mapped to columns in the appropriate tables. This is an example of object/relational mapping (ORM). Aggregation [7] is a stronger form of association. It represents the "has-a" or "part-of" relationship. On fig. 2. is shown the scheme of the design process. The domain model incorporates the knowledge resources and allows knowledge management and knowledge reasoning. Data objects are stored in tables. Following the UML class diagram as well-known modeling tool [12] useful environment and efficiency of modeling are achieved. The model development process is provided with class diagram or entity relationship diagram (on fig. 3), from which automated transformation into object model is generated. The visual modeling environment is an intuitive way for object models and serves as resource-centric interface for decisions assisting.

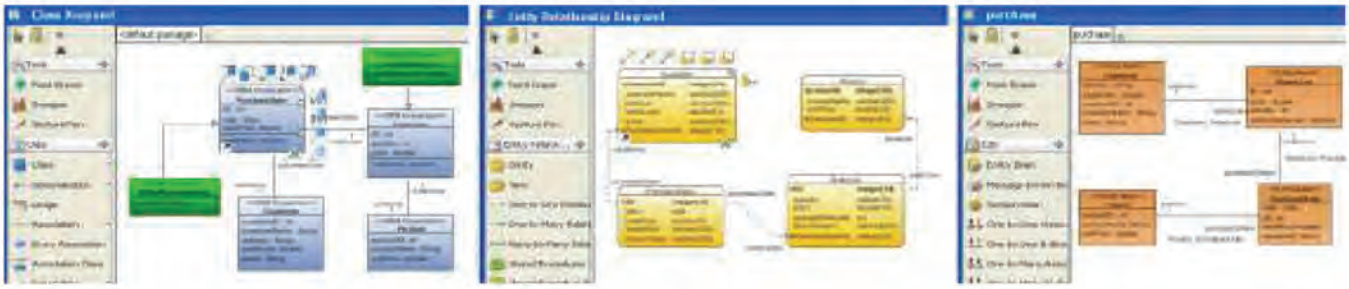

Figure 3. User interface within Visual Paradigm.

The aim is to achieve fast and simple learning models that result in small rule bases, which can be interpreted easily. The wide used support vector [3], [9] machines (SVM) are effective tool for optimal classifier capacity tailored on the given task problem. The SVM success depends on the tuning of several parameters, which affect the generalization error. Another possible model is Adaptive-Network-based Fuzzy Inference Systems (ANFIS) as hybrid learning method. Its limit is the exponentially growing number of fuzzy rules. The very fast and simple decision tree model is also applied in this investigation. In the context of medical diagnosis, the extraction of statistically independent components has been proposed as an objective of early sensory processing, finding the important attributes and primary dependencies.

There is a scope of many diseases with the same symptoms, and also the degree or level of the symptoms is absent from the knowledge. To overcome the first problem, the knowledge engineer should design the knowledge base with more specific rules. For the second kind of data modelling problem further information needs to be gathered and integrated in domain model. Several mathematical models were developed in order to optimize the information provided by the sensor network.

IV. Experimental Results

Nowadays it is accepted that there are no algorithms able to make exact classifications in a general domain. Local information is extracted from features associated to each local structure and provides information about its type of structure. Contextual information is incorporated by taking into account the relative spatial distribution of these local features. Finally global information is obtained as a result of considering the local features over neighborhoods of the feature set [13].

In this work we are particularly interested in studying the medical signal measurements and diagnosing process of heart diseases in Bulgaria and in other countries. First we analyze the public available patients' database [10] for heart attack and evaluate three different models. Then we compare the predicted outcome for new patients based on the model accuracy. The aim is to propose a guided assistance and to determine which attributes (medical check-up/ blood tests) are important in choosing predictors (measurements).

A. SVM modelling

The SVM can be used to learn highly accurate models from data. It is based on quadratic optimization [11] of convex function. This is realized by nonlinear mapping using so-called kernel functions. Let **x** be recorded data, the outcomes are 2 classes: C_1 for positive diagnosis, and C_2 – for negative. Via normalizing any uniformly separating hyperplane can be defined by the property:

$\mathbf{w}^T\mathbf{x}+b \geq 1$, when $\mathbf{x}$ is in class C_1 and

$\mathbf{w}^T\mathbf{x}+b \leq -1$ for $\mathbf{x} \in C_2$ (class C_2) (1)

By maximizing that distance we minimize the following dot product ½ $\mathbf{w}^T\mathbf{w}$. This represents the quadratic constrained optimization problem.

Recent developments in the literature on the SVM and other kernel methods [4], [5] have emphasized the need to consider multiple kernels, or parameterizations of kernels, and not a single fixed kernel. This provides the needed flexibility and also reflects the fact that practical learning problems often involve multiple, heterogeneous data sources.

The results from SVM model are obtained from SPIDER MatLab® toolbox and the decision boundary for the binary diagnosis from heart dataset is presented on fig. 4. All results are obtained with stopping tolerance $\varepsilon = 10^{-6}$.

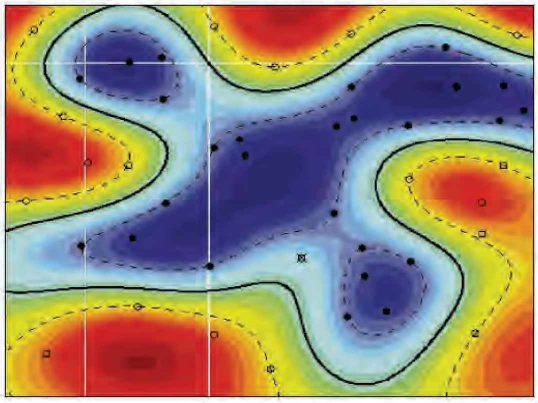

Figure 4. Visualization of SVM binary decision for the conducted experiment with RBF kernel, parameter $\alpha = 0,3$.

The concept of a kernel mapping function is very powerful. It allows SVM models to perform separations even with very complex boundaries shown on fig.4. The prediction accuracy tends to 96% and it is efficient and roughly scales as $O(n.d_{max}{}^2)$ where n is the number of training examples. The number of basis functions is usually far less than the number of SVM support vectors.

B. ANFIS modelling

The Adaptive-Network-based Fuzzy Inference Systems (ANFIS) are proposed b Roger Jang in 1993. As a part of MatLab® toolbox ANFIS applies two techniques in updating parameters. This approach is called hybrid learning method.

We can then rank the importance of the input variables according to the range covered by their fuzzy curves. If the fuzzy curve for a given input is flat, then this input has a little influence is not a significant input. If the range of a fuzzy curve is about the range of the output data, then it is the most important to the output variable. For the conducted experiments the resulting decision surface is given on fig. 5. It can be seen that on the upper left corner there exists a linear dependence between the 3 selected attributes, which gives a good separating decision. For the x3 attribute in the range [1,2] "class outcome" is non clearly distinguished, so we need more rules.

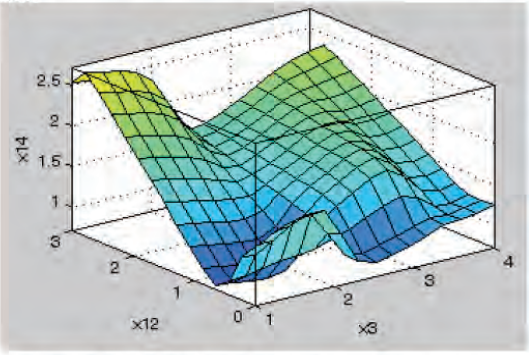

Figure 5. ANFIS modelling decision surface.

When using this model with automatically extracted final rule-base the achieved accuracy is 0.4942 as shown on fig. 6. Usually the generated rules enumerate all possible combinations of membership functions of all inputs. This leads to an exponential explosion even when the number of inputs is 13. This makes the ANFIS model not very useful for the conducted experiments. The precision of the derived model coincides with the decision tree model and serves in our experiments as confirmation of the correct decision.

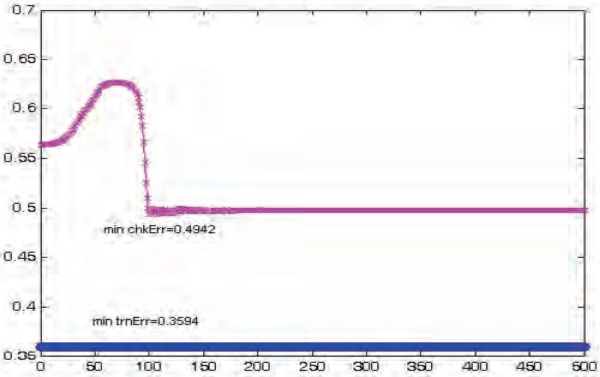

Figure 6. The accuracy of ANFIS model for more than 100 training epoch is about 0,5 (depicted in pink curve)

C. WEKA model

To test the above described method we conduct several experiments in Weka [11]. The multidimensional models are also created and their associations are used in specific applications. This is rather useful for transferring a classifier setup from the Weka Explorer over to the Experimenter without having to setup the classifier from scratch.

For the first experiment we run decision tree classifier to find the most important attribute. The result is shown on fig. 7. The tree validation method is Out of Bag (OOB), and 3 predictors (out of 13) were used for each split. The maximum depth of any tree in the forest is 17. The importance is:

```
==== Overall Importance of Variables ===
      Variable   Importance
        a13        100.000
        a12         52.522
        atr3        23.355
```

and the minimum error found by search is 0.13378. This result is then used to test the accordance with other models.

To determine the predicted value we start with the root node and based on the value of the splitting variable, proceed analysis to child node. The value of the target variable shown in the leaf node is the predicted value – in our case this is 3.4 for attribute 13. One can use the tree to make inferences that help to understand the model.

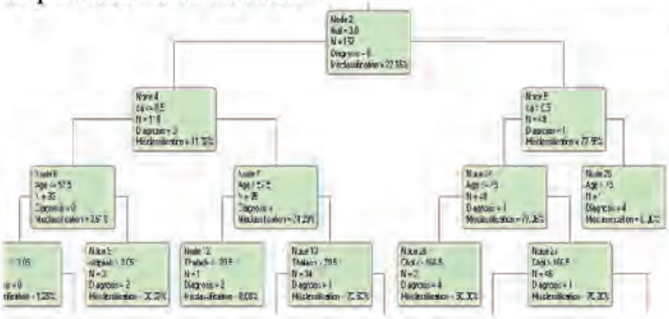

Figure 7. Decision tree result for the heart attack diagnosis.

This is one of the great advantages of decision trees over classical regression and neural networks – decision trees are easy to interpret even by non-technical people.

D. Comparative precision and discussion

In diagnosis applications the outcome may be the prediction of disease vs. normal or in prognosis applications. In this study we have used three different modeling designs to evaluate the influence of these techniques for data analysis.

Separating the classes with a large margin minimizes the bound on the expected generalization error. Unlike other algorithms, the SVM makes no assumptions about the relationships between a set of features (attributes). This allows us to identify and determine the most relevant features used in a model and their dependencies.

The ANFIS model shows a better accuracy when specific rules are chosen, but it becomes computationally infeasible on large data sets. To reduce the time and space complexities, a popular technique is to obtain low-rank approximations, by using greedy approximation. Applying the architectural support paradigm provide a possibility to understand where artificial intelligence technologies offer potentials to optimize the interaction of human operators.

Our experience with these software applications indicates that near-optimal solutions are often good enough in practical applications. By observing practical SVM implementations only approximate the optimal solution by an iterative strategy, self developed kernel methods may be applied and scaled up by exploiting such ‘approximateness’. We will try different stopping condition in order to avoid unstable behaviors. We aim at understanding what influences the SVM accuracy.

V. Conclusion

The considered experimental methodology is applied to the explored data and the results from the three models serve as a demonstration that correct decision is produced. They allow to group models into hierarchies to create a simplified view of components. High-level information is presented clearly and concisely, while detailed information is easily hidden in the model hierarchy. It is a hybrid model that simultaneously addresses fast and slow medical processes (single heart beat) that are implemented in mixed discrete and continuous modes.

The problem today is that there are not enough trained human analysts available who are skilled at translating all of measured data into knowledge. The ongoing remarkable growth in the field of data mining and knowledge discovery has been fueled by a fortunate confluence of a variety of factors: explosive growth in data collection; storing in data warehouses with access to current database; increased access from Web navigation; globalized economy; growth in computing power and storage capacity. Therefore new architectural support is needed for the design of data models and better visualization.

Acknowledgement

This research is supported by the NSRF of the Bulgarian Ministry of Education and Science as part of I- № 1406/2004.

References

[1] P. Andreeva, “Data Modelling and Specific Rule Generation via Data Mining Techniques”, *Proc. of, CompSysTech’06,* Bulgaria, 15-16 June 2006, pp. III.A 17-23.

[2] P. Brito, Bertarand P., Cucumel G., De Carvalho F. (eds) *Studies in classification, data analysis and knowledge organization*, Selected Contributions in Data Analysis and Classification, Springer Verlag, 2007.

[3] P. Bartlett, J. Sh-Taylor, “Generalization performance of support vector machines and other pattern classifiers”, in: B. Schölkopf, C. Burges, and A. Smola, (Ed.): *Advances in Kernel Methods*, MIT Press, MA 43-54, 2002.

[4] Canu S., Y. Grandvalet, “More Efficiency in Multiple Kernel Learning”, *Proceedings of the 24 th International Conference on Machine Learning, Corvallis*, OR, 2007.

[5] Chapelle, O., “On multiple kernel learning”, *Machine Learning*, **46**(1-3):131–159, 2005.

[6] L. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms,* Wiley, 2004.

[7] A. Rozeva, “Implementation of Aggregations in a Data Warehouse Logical Scheme– Framework and Mechanism”, *Third Int. Scientific Conference ‘Computer Science’2006’*, Istanbul, Turkey, Oct.12-15 2006, vol. II, pp. 330-335.

[8] S. Russell and Norvig, P., *Artificial Intelligence: a modern approach.* Prentice-Hall, 2nd edition, 2003.

[9] D.M.J. Tax and P. Laskov, “Online SVM learning: from classification to data description and back”, in C. et al. Molina, editor, *Proc. NNSP*, pp. 499–508, 2003.

[10] UCI Repository, http://archive.ics.uci.edu/ml/

[11] University of Waikato, ML Program WEKA, www.cs.waikato.ac.nz/ml/weka (2007)

[12] Visual Paradigm for the Unified Modeling Language, (accessed 2008) http://www.visual-paradigm.com

[13] S. Vassileva, G. Georgiev, S. Mileva, “Knowledge-Based Control Systems via Internet”, Part I. *Appl. in Biotechnology Bioautomation*, ISSN 1312-451X, pp. 37-48, 2005.