A Hybrid Data Mining and Case-Based Reasoning User Modeling System Architecture

Dr. Dino Isa, Dr. Peter Blanchfield, and Chen Zhi Yuan

Abstract—**In this paper we present the architecture of a hybrid Data Mining and Case-Based Reasoning system which incorporates a user model to help filter information in order to make it more relevant to the user. The main issue of implementing this hybrid system is a knowledge base which is first derived from a domain information database and where a priori user information is unavailable, the system builds a user preference table by monitoring mouse clicks and updating or refining a rule based user model. The user model and knowledge base are classified using a support vector machine and retrieved from the case based reasoning cycle using a self organizing map. The objective of this project is to combine data mining technology and artificial intelligence pattern classifiers as a means to construct a user oriented Knowledge Base and to link this to the case-based reasoning cycle in order to provide domain specific user relevant information to the user in a timely manner.**

Index Terms—**Data Mining, Artificial Intelligence, Case-Based Reasoning, Knowledge Base.**

I. Introduction

The problem faced by many organizations today is not one of too little data; it is the exact opposite [1]. To alleviate this problem of too much data (or more precisely, irrelevant data), one may resort to many different techniques that will enable the classification of information which is specifically useful only to certain classes of users. However, caution must be exercised in these instances as there many sources of error which can make a system misclassify information rendering the entire knowledge base useless. As such, hybrid systems combining case-based reasoning [2], data mining [3] and artificial intelligence [4] have emerged to address issues related to the building of a user relevant knowledge base; the classification of new information to be added to the knowledge base and the retrieval of information most relevant to a user query is also addressed by these new hybrid systems. In this paper we present a novel architecture of a hybrid Data Mining and Case-Based Reasoning User modeling system.

The rest of this paper is organized as follows: Section 2 presents objectives and related techniques. Section 3 describes in detail the architecture of the hybrid system. Section 4 presents the present work. The conclusion is discussed in section 5.

II. OBJECTIVES AND FOUNDATION

The design of flexible and adaptable user oriented hybrid systems is a very complex task implying the sequential ordering many software components and algorithms. These components correspond to data vectorization step of a classification process, going from low level data mining processes to high level artificial intelligence techniques. Many domain specific system such as user modeling systems or artificial intelligence hybrid systems have been described in literature [5] [6] [7]. Even when the applied strategies are designed as generic as possible, the illustration given for the system are limited to the text document and, moreover do not develop any vectorizing algorithm to quantitate the input raw data set.

Actually, to the best of our knowledge, no such complete and generic system exists because of the necessity to have an excellent know-how in the implementation of a hybrid intelligent system. Many knowledge categories are involved in this expertise, and that a good manner to let the hybrid system to be as intelligent as possible is to vectorize the raw data set into multidimensional vector in order to fulfill the requirement of implementing artificial intelligence techniques.

From the end user's point of view, this hybrid system appears to be an intelligent user oriented system, allowing to modify raw data set and to search the latest analysis result on the base of the user interface component. From the point of view of the hybrid system developer, the platform has allowed us to interface all the process units, such as data mining process, user modeling process, case-based reasoning process.

The platform relies on three specific concepts:

1) Dynamic model construction: The principle is to allow the user to interact with the system for the development of the user model and domain model, consisting in the data mining process unit. Furthermore, running a scenario collect user preference, after be transformed into vector which becomes part of the user model.
2) Vectorization approach: All vectors that are extracted from domain database and user database are achieved in the data mining process unit by applying a novel algorithm. It evolves two functions considering continuous column and discrete column. Thus the vector can be viewed as a communication channel between real world documents and artificial intelligence signals. Such an approach avoids the problem of intelligence

Manuscript received April 19, 2008.

Dr. Dino Isa is with the Engineering and Computer Science Faculty, the University of Nottingham Malaysia Campus (phone: 006-03-8924-8116; fax: 006-03-8924-8017; e-mail: dino.isa@nottingham.edu.my).

Dr. Peter Blanchfield is with the Computer Science School, University of Nottingham UK Campus, (e-mail: author@lamar. colostate.edu).

Chen Zhi Yuan is a Ph.D student under the Engineering and Computer Science Faculty, the University of Nottingham Malaysia Campus, (e-mail: eyx6czy@nottingham.edu.my).

techniques can not deal with text as input signal.

3) A plug-in oriented architecture: The principal is to permit the integration of heterogeneous software component. As a consequence, the developer can conveniently add new processing unit, thus make the hybrid system easier to upgrade.

Actually, this hybrid system does not aim at representing all categories of knowledge that are implicitly involved, but offer major domain information. In the following section we will present the architecture of the hybrid system.

III. HYBRID SYSTEM ARCHITECTURE

A. *Architecture Overview*

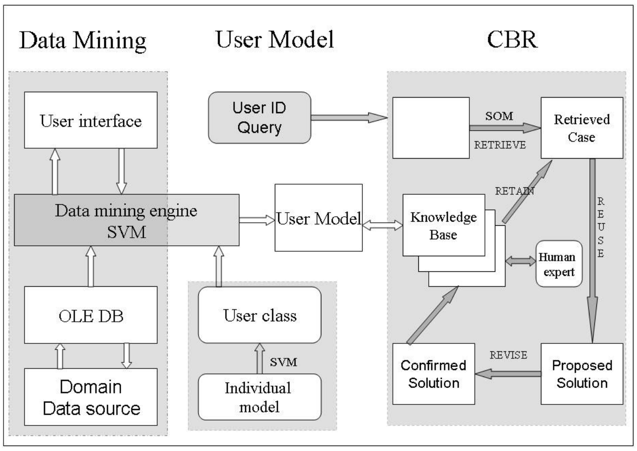


Figure 1. The architecture of the system

The concepts presented in section 2 have led to the design of a hybrid Data Mining and Cased-based Reasoning User Modeling system. The architecture is illustrated in Figure 1. The hybrid system contains five main components:

1) Individual model, comparable to the blackboard containing the current user's related character information.
2) A set of Component Object Model (COM) [8] interfaces that provide applications with uniform access to data stored in diverse information sources.
3) A data mining engine which classified both user class and domain information vectors.
4) A knowledge base, containing the representation of classified user information and combined with interested domain knowledge.
5) A problem-solving life-cycle called case-based reasoning cycle, assisting in retrieve reuse revise and retain the knowledge base.

B. *Vectorization*

Our project is focused on user which can be considered as a set of object. The user model centralized the major part of the hybrid system and is progressively enriched by the data mining process. In order to classify individual model into user class the vectorization method are applied. We defined two types of column in order to standardize the raw data table. The schema of the algorithm is specified in Figure 2 which derives the numeric vector by implementing different functions. The schema is not exhaustive and can evolve with new data, according to user need.

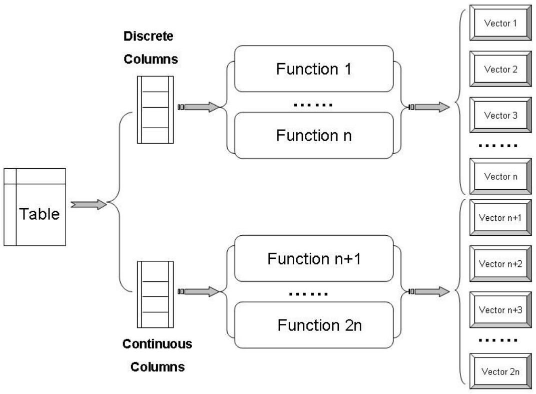


Figure 2. The schema of the vectorization algorithm

The next sub-section shows the main issue for classification task in data mining process.

C. *Classification*

The data mining process contains four components: User interface, Data mining engine, OLEDB, Domain data source. The goal of this sub-section is not to describe the whole procedure of this process, but to illustrate how to classify vectorized data set.

In order to use support vector machines and avoid the complex and subtle training algorithm, a SVM learning algorithm which is called Sequential Minimal Optimization (or SMO) [9] is implemented.

Table 1. Timings of algorithms for various experiments

Training Set Size	SMO Time (sec)	Decomposition Time (sec)	Chunking Time (sec)
11221	13.7	217.9	20711.3
11221	21.9	n/a	21141.1
11221	339.9	3980.8	17164.7
11221	523.3	737.5	n/a
11221	1433.0	n/a	14740.4
49749	1810.2	n/a	n/a
49749	2477.9	2949.5	23877.6
49749	2538.0	6923.5	n/a
49749	4589.1	n/a	17332.8
60000	19387.9	38452.3	33109.0
49749	23365.3	n/a	50371.9
49749	24758.0	n/a	n/a

Support Vector Machine (SVM) is a classification technique that has received considerable attention. Promising empirical results have shown in many practical applications, from handwritten digit recognition to text categorization. SVM also works very well with high-dimensional data and avoids the curse of dimensionality problem. To facilitate effective and efficient classification in the knowledge base creating process SVM is adopted in our hybrid system. Due to the QP [10] problem that arises from SVMs can not be solved via standard QP techniques, we import SMO algorithm. Sequential Minimal Optimization quickly solves the SVM QP problem without using numerical QP optimization step at all. It decomposes the overall QP problem into fixed size QP sub-problems. Unlike previous methods "chunking" [11] or decomposition techniques [12], however, SMO chooses to solve the smallest possible optimization problem at each step. In [9] the SMO algorithm is test against the standard chunking algorithm and against the decomposition method on a series of benchmarks.

As can be seen in Table 1, standard chunking is slower than SMO for the data sets shown, although Decomposition has the advantage over standard chunking, SMO is still the fastest one among these three algorithms.

D. *Case-Based Reasoning Cycle*

The notion of case-based reasoning (or CBR) cycle is another main issue of our hybrid system. According to Aamodt and Plaza [2] a CBR cycle is consists of four parts:

1) RETRIEVE the most similar case(s);
2) REUSE information and knowledge stored in the case to solve the problem;
3) REVISE the proposed solution if necessary, and
4) RETAIN the new solution as a part of a new case for future problem solving.

Within the CBR process the new query from the current user will be evaluated to find out the most relevant case base that is the domain specific knowledge which the user is favorite to, and then recommend it to the current user. We plan to perform case classification with trained SOM [13] network to enlighten the work of recommendation. The SOM is one of the most distinguished unsupervised learning algorithms. The principal goal of why we are using SOM is to transform arbitrary user cases into a discrete map in order to group similar cases from the knowledge base. The SOM technique is related to vector quantization which is a data compression technique, the input vectors is divided into a number of distinct regions, and for each region a reconstruction vector is defined. The collection of all these reconstruction vectors constitutes is called the "code book" of the vector quantizer. A vector quantizer with the minimum encoding distortion is called a Voronoi or nearest neighbor quantizer. The SOM provides an approximate method for computing the Voronoi quantizer in an unsupervised manner using competitive learning. The detailed steps for implementing the SOM can be described as: the computation of the feature map is the first stage; and then at the second stage fine-tunes the SOM by using the class information to move the code book vectors slightly for improving the quality of the classifier decision regions.

E. *User Interface*

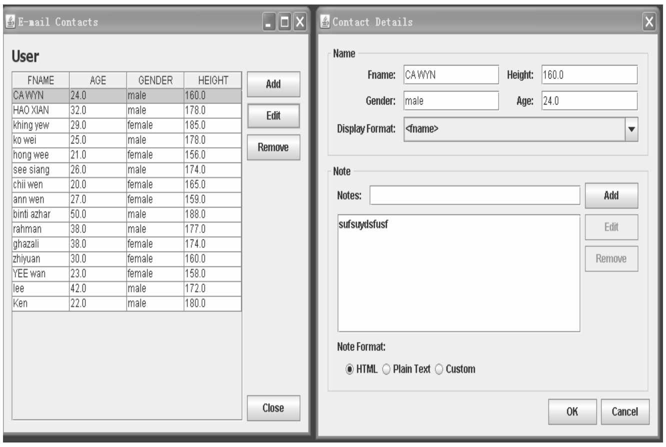
Two user interfaces have been integrated into this hybrid system. The first one, called UMI (User Modification Interface), aims at providing user a friendly interface to modify the domain and individual model database. The second one, called UQI (User query interface) is a framework for querying and interacting with the knowledge base. Although the two interfaces can be used separately, they communicate together, allow the user to watch, correct and tune immediately of the Knowledge Base construction process.

1) The UMI allows the user to modify the domain database and individual model. For a given table in the raw database the UMI is able to list the column content which may be contained. After the user choose a unit, the UMI tool supplies the table sheet so as to be able to launch the correspond process. The UMI has three modes for modification execution:
   - □ ADD mode
   - □ EDIT mode
   - □ REMOVE mode
2) The UQI is a framework for cooperative and interactive with the Knowledge Base. It offers an interface that uses users' query to be a new problem to interact with the Knowledge Base.

IV. PRESENT WORK

Our hybrid system is a multi process platform, so that different procedure can be developed in a parallel manner. General speaking our present work can be divided into four aspects; one is the interface development (a simple demo is shown in figure 3), in the second step we concentrates on the vectorization task which has been presented in our previous paper [14], the third component is the classification part, that is the ongoing works, and the last procedure CBR combined with SOM is our future plan.

A. *UMI*



E-mail Contacts

User

FNAME	AGE	GENDER	HEIGHT
CA WYN	24.0	male	160.0
HAO XIAN	32.0	male	178.0
khing yew	29.0	female	185.0
ko wei	25.0	male	178.0
hong wee	21.0	female	156.0
see siang	26.0	male	174.0
chii wen	20.0	female	165.0
ann wen	27.0	female	159.0
binti azhar	50.0	male	188.0
rahman	38.0	male	177.0
ghazali	38.0	female	174.0
zhiyuan	30.0	female	160.0
YEE wan	23.0	female	158.0
lee	42.0	male	172.0
Ken	22.0	male	180.0

Contact Details

Name
Fname: CA WYN
Height: 160.0
Gender: male
Age: 24.0
Display Format: <fname>

Note
Notes:
sufsuydsfusf

Note Format: ● HTML ○ Plain Text ○ Custom

Figure 3. User modification interface

The demo shown in Figure 3 is developed under java NetBeans IDE, in this application the individual model is represented by Java DB (Derby) database, the database server is the Sun Java System Application Server. In the real world we are not limited to working with Derby databases, to work with other databases, we need to install the database server and the JDBC driver. For the IDE to communicate with a database server, the IDE requires a driver supporting the JDBC API ("JDBC driver"), which translates JDBC calls into the network protocols.

B. SMO

There are three components to SMO: an analytic method to solve for the two Lagrange multipliers, a heuristic for choosing which multipliers to optimize, and a method for computing b.

1) An analytic method: In this step SMO first computes the constraints on these multipliers and then solves for the constrained maximum. The reason for why two is the minimum number of Lagrange multipliers that can be optimized is: if SMO optimized only one multiplier, it could not fulfill the linear equality constraint at every step. Because there are only two multipliers, the constraints can easily be displayed in two dimensions. The bound constraints cause the Lagrange multipliers to lie within a box, the linear equality constraints causes the Lagrange multipliers to lie on a diagonal line. Thus, the constraint maximum of the objective function must lie on a diagonal line segment.
2) Heuristics for choosing which multipliers to optimize: In order to speed convergence, SMO uses heuristics to choose which two Lagrange multipliers to jointly optimize. SMO will always optimize two Lagrange multipliers at every step, with one of the Lagrange multipliers having previous violated the KKT conditions before the step, that is, SMO will always alter two Lagrange multipliers to move uphill in the objective function projected into the one-dimensional feasible subspace. SMO will also always maintain a feasible Lagrange multiplier vector. Therefore, the overall objective function will increase at every step and the algorithm will converge asymptotically.
3) Threshold: The above two steps have solved the Lagrange multipliers, but have not determined the threshold b of the SVM, so b must be computed separately. After each step, b is re-computed, so that the KKT conditions are fulfilled for both optimized examples. A cached error value E is kept for every example whose Lagrange multiplier is neither zero nor C. When an error E is required by SMO, it will look up the error in the error cache if the corresponding Lagrange multiplier is not at bound. Otherwise, it will evaluate the current SVM decision function based on the current vector.

V. CONCLUSIONS

The proposed hybrid Data Mining and Case-Based Reasoning User Modeling system is a multi purpose platform and is characterized by three major processes. At first its architecture relies on individual models and a domain database, vectorization processing unit communicate through the raw data set, such an approach avoid the data inconsistency usually met in classifying documents chain when implement artificial intelligence tools. Secondly the framework is based on a plug-in oriented architecture. Developers can conventionally add new components, thus making the system upgrade easily. A fast training SVM algorithm SMO is also applied, so that SVM can be implemented easily. Thirdly the hybrid system is equipped a CBR cycle which is facilitated by the self-organizing map. By running this cycle, the query from the user will be looked as a new case or a new problem then search the most similar solution in the Knowledge Base, and if there is not a similar solution, this new query can be stored as a new case. Thus the architecture is a real modular because a user can create his own cases, integrate his own case into the Knowledge Base.

More than a software environment, the hybrid system must be considered as a general tool for integrating various artificial intelligence tools and system.

REFERENCES

[1] JIM LEE, “Data Explosion”, *Disaster Recovery Journal*, fall 2004, Volume 17, Issue 4.

[2] Aamodt, A. and Plaza, E., “Case-based reasoning: foundational issues, Methodological variations, and system approaches”, *AI communications*, 7(1), 1994, pp. 39-59.

[3] Usama Fayyad, G. Paitetsky-Shapiro, and Padhrais Smith, “knowledge discovery and data mining: Towards a unifying framework”, *proceedings of the International Conference on Knowledge Discovery and Data Mining*, 1996, pp. 82-22.

[4] Stuart J. Russell and Peter Norvig, *Artificial Intelligence A Modern Approach*, Prentice-Hall International Inc, 1995.

[5] Vassileva, J., "A practical architecture for user modeling in a hypermedia-based information system", *Proceedings of Fourth International Conference on User Modeling*, Hyannis, MA, August 1994, pp 15-19.

[6] Vadim I. Chepegin, Lora Aroyo, Paul De Bra, “Ontology-driven User Modeling for Modular User Adaptive Systems”, *LWA*, 2004, pp.17-19.

[7] Watson, I, *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, Morgan Kaufmann Publishers, Inc., San Francisco, CA, 1997.

[8] ZhaoHui Tang, Jamie MacLennan, *Data Mining with SQL server 2005*, John Wiley & Sons, 2005.

[9] J. C. Platt., “Fast training of SVMs using sequential minimal optimization”, *Advances in Kernel Methods- Support Vector Learning*, 1998, pp185-208.

[10] L. Kaufman, “Solving the quadratic programming problem arising in support vector classification”, *Advances in Kernel Methods- Support Vector Learning,* 1998, pp147-168.

[11] V. Vapnik, *Estimation of Dependences Based on Empirical Data*, Springer-Verlag, 1982.

[12] E. Osuna, R. Freund, and F. Girosi, “Improved training algorithm for support vector machine”, *IEEE Neural Networks in Signal Processing 97*,1997.

[13] F. Murtagh. Interpreting the Kohonen, “self-organizing map using contiguity-constrained clustering.”, *Pattern Recognition Letters, 1995, pp. 399–408.*

[14] Chen Zhi Yuan, Dino Isa, Peter Blanchifield, “Data preprocessing in a hybrid system”, *proceedings of the third Malaysia Software Engineering Conference,* 2007, pp332-336.