

PANDA Technical Report Series

TR Number: **PANDA-TR-2003-04**

Title: *Recent Advances on Pattern Representation and Management*

Author(s): M. Vazirgiannis (AUEB), M. Halkidi (AUEB), D.A. Keim (Konstanz), I. Ntoutsis (CTI/Piraeus), A. Pikrakis (Athens), S. Theodoridis (Athens), Y. Theodoridis (CTI/Piraeus), G. Tsatsaronis (AUEB), E. Vrachnos (AUEB)

Date: 30 December 2003



Research supported by the Commission of the European Communities under the Information Society Technologies (IST) Programme – Future and Emerging Technologies (FET)

Recent Advances on Pattern Representation and Management

Michalis Vazirgiannis^{1*}, Maria Halkidi¹, Daniel A. Keim², Irene Ntoutsi^{3,4},
Aggelos Pikrakis⁵, Sergios Theodoridis⁵, Yannis Theodoridis^{3,4},
George Tsatsaronis¹, Euripides Vrachnos¹

¹ Dept. of Informatics, Athens University of Economics and Business, Greece

² Dept. of Computer Science, University of Konstanz, Germany

³ Computer Technology Institute, Greece

⁴ Dept. of Informatics, University of Piraeus, Greece

⁵ Dept. of Informatics and Telecommunications, University of Athens, Greece

ABSTRACT: Data intensive applications produce complex information that is posing requirements for novel Database Management Systems (DBMSs). Such information is characterized by its huge volume of data and by its diversity and complexity, since the data processing methods such as *pattern recognition*, *data mining* and *knowledge extraction* result in *knowledge artifacts* like clusters, association rules, decision trees and others. These artifacts, called *patterns*, need to be stored and retrieved effectively and efficiently. In this paper, we review the concept of patterns and their applicability in several research domains and we define the knowledge domain related to the *PANDA* project. We examine the different types of patterns that are extracted from a data set, in order to gather the necessary requirements for the definition of a pattern model. This model will constitute the heart of the Pattern Base Management System that will be designed.

KEYWORDS: patterns, data mining, pattern modeling, pattern-bases, information retrieval, Pattern Base Management Systems

(*) Contact Author. E-mail: mvazirg@aueb.gr

1. Introduction - Motivation

With the advent of hardware advances, complex information resulting from data intensive applications is posing requirements for novel Database Management Systems (DBMSs). Such information possesses a number of key features such as:

- *Huge volume of data*: For example, images being collected daily from various sources (satellites, etc.) that need to be stored and retrieved efficiently; in this case data can be represented concisely by a set of patterns (for instance, a mathematical formula might represent the trajectory of a satellite). Moreover, huge traditional databases are growing due to extensive transaction paces in various domains (banking, stock exchange, telecommunication etc. databases).
- *Diversity and complexity*: Data processing methods (pattern recognition, data mining, knowledge extraction) result in knowledge artifacts (i.e., clusters, rules, *patterns* in general) that need to be as well managed by a DBMS-like environment.

It is obvious then that the knowledge artifacts arise as significant representational primitives in recently computerized application domain and therefore call for integrated and efficient DBMS support. Unfortunately patterns have not been treated as persistent objects that can be stored, retrieved and queried. It is now the time to tackle the challenge of integration between the two fields (pattern and data) by designing fundamental approaches for providing database support to patterns.

Various application domains dealing with patterns (telecom, medical, environmental information systems, etc.) will directly benefit from a system that integrates data and pattern management. This will be due to the fact that database support will enhance the maintenance and manipulation of both their data collections and artifacts produced in the form of patterns. Another field of advance results from the fundamentally novel paradigm arising from patterns and affecting various database research areas, such as data models, query languages, query processing and indexing techniques, visual user interfaces.

In this paper, we review the concept of patterns and their applicability in several research domains related with the proposed work and define the knowledge domain related with the PANDA project [PANDA]. It is important that we interrelate these domains in order to be

able to define the problem in comprehensive and complete way and come up with requirements on how a management system for patterns should be.

The remainder of the paper is organized as follows. In Section 2 a review of patterns application fields and existing research results for patterns is presented. There is a rich domain of fundamental research related to patterns in mathematics, data mining, pattern recognition as well as in several application domains. Section 3 follows discussing the current issues in modeling Data Mining processes. The innovation of a pattern management system which is the subject of the PANDA project is presented in Section 4 along with a set of requirements for designing a system that can handle modeling, storage, visualization and retrieval of patterns.

2. Application fields and pattern usage

Various application domains related with data management (storage, process, retrieval, data analysis) result in different forms of patterns representing the data insights. In the sequel we present some representative pattern application domains and the corresponding types of patterns they produce.

2.1. Patterns in Data Mining

The last decade has brought an explosive growth in our capabilities to both generate and collect data. Advances in database technology have provided us with the basic tools and methods for efficient data collection, storage and lookup of datasets. The result is that a flood of data has been generated and a growing data glut problem has been brought to the worlds of science, business. Also our ability to analyze, interpret large bodies of data and extract "useful" knowledge has outpaced and the need for new generation of tools and techniques for intelligent database analysis has been created. This need has been recognized by researchers in different areas (artificial intelligence, statistics, data warehousing, on-line analysis processing, expert systems and data visualization) and a new research area is emerged, known as *Data Mining*.

Data Mining is mainly concerned with methodologies for extracting data patterns from large data repositories. The extracted patterns are evaluated based on some interestingness measures that identify patterns representing knowledge, i.e., *interesting patterns*. These patterns are presented to the user and may be stored as new knowledge in the knowledge base. Then, we could adopt a broader view of data mining functionality, considering data mining as

the process of discovering interesting knowledge from large amounts of data stored in databases, data warehouses or other information repositories.

There are many data mining algorithms that accomplishing a limited set of tasks produce a particular enumeration of patterns over data sets. These main tasks, according to well established data mining process [BL96], are: i) the definition/extraction of clusters that provide a classification scheme, ii) the classification of database values into the categories defined, ii) the extraction of association rules or other knowledge artefacts, iii) discovery and analysis of sequences.

Since there are various types of data stores and database systems on which data mining tasks can be performed, different kinds of data patterns can be mined. In some cases users have no idea about the kinds of patterns in their data that could be interesting. Thus it is important a data mining system to mine and store multiple kinds of patterns so as to accommodate different user expectations and applications. Another requirement in data mining is the granularity of data mining results. There are cases that it is important to have different levels of abstraction for the patterns mined from a data repository depending on the application or user requirements.

In the sequel, we discuss the data mining functionalities and the kinds of patterns that can be mined from an amount of data.

2.1.1 Clustering

Clustering is the process of partitioning a given data set into groups (clusters) such that the data points in a cluster are more similar to each other than points in different clusters [GRS98]. The clustering process may result in different partitions of a data set, depending on the criteria used for clustering. Thus, there is a need of pre-processing before we apply a clustering task in a data set. The basic steps of a clustering process can be summarized as follows [FSSU96]:

- *Feature selection.* Clustering is performed on certain features.
- *Clustering algorithm.* The clustering algorithm is characterized by a proximity measure and a clustering criterion. The correctness of a clustering algorithm results is verified using appropriate criteria and techniques.

Cluster analysis is a major tool in a number of applications in many fields of business and science. Hereby, we summarize the basic directions in which clustering is used [TK98]:

- *Data reduction.* Clustering can be used to partition the data set into a number of “interesting” clusters. Then, instead of processing the data set as an entity, we adopt the representatives of the defined clusters in our process. Thus, data compression is achieved.
- *Hypothesis generation.* Cluster analysis is used here in order to infer some hypotheses concerning the data.
- *Hypothesis testing.* In this case, cluster analysis is used for the verification of the validity of a specific hypothesis. This is done by applying a clustering algorithm to a representative set of the data.
- *Prediction based on groups.* In that case new items can be assigned to a specific cluster given the relation between their properties and the properties of the patterns of the cluster.

More specifically clustering can be applied [HK01] in the fields of *Business, Biology, Spatial data analysis, Web mining* and others.

In general terms, clustering may serve as a pre-processing step for other algorithms, such as classification, which would then operate on the detected clusters.

2.1.2 Classification – Decision Making

The classification problem has been studied extensively in statistics, pattern recognition and machine learning community as a possible solution to the knowledge acquisition or knowledge extraction problem [RS98]. A number of classification techniques have been developed and are available in bibliography. Among these, the most popular are: *Bayesian classification, Neural Networks* and *Decision Trees*.

Bayesian classification is based on Bayesian statistical classification theory. The aim is to classify a sample x to one of the given classes c_1, c_2, \dots, c_N using a probability model defined according to Bayes theory [CS96]. Each category is characterized by a prior probability of observing facts that belong to the category c_i . An input pattern is classified into a category with the highest posterior probability.

Decision trees are one of the widely used techniques for classification and prediction. A number of popular classifiers construct decision trees to generate classification models.

A decision tree is constructed based on a training set of pre-classified data. Each internal node of the decision tree specifies a test of an attribute of the instance and each branch descending of that node corresponds to one of the possible values for this attribute. Also, each leaf corresponds to one of the defined classes. The procedure to classify a new instance using

a decision tree is as follows: starting at the root of the tree and testing the attribute specified by this node; successive internal nodes are visited until a leaf is reached. At each internal node, the test of the node is applied to the instance. The outcome of this test at an internal node determines the branch traversed and the next node visited [Mit+97]. The class for the instance is the class of the final leaf node.

Another classification approach used in many data mining applications for prediction and classification is based on neural networks. More specifically, the methods of this approach use neural networks to build a model for classification or prediction. The main steps for this process (i.e. building a classification model) can be found in [BL97].

As we have already discussed classification is a form of data analysis that can be used to define data models describing data classes or predict data trends. It can be used for making intelligent bases decisions in business and science.

2.1.3 Association Rules

Association rules reveal underlying interactions between attributes in the data set. These interactions can be presented with the following form: $A \rightarrow B$, where A, B refer to sets of attributes' values in underlying data. More specifically, A and B are selected so as to be frequent item sets. A formal statement of the problem can be found in [AS94].

The intuitive meaning of such a rule is that records in the dataset, which contain the attributes in A, tend also to contain the attributes in B [SA95]. We note also that the extracted rules have to satisfy some user-defined thresholds related with association rules measures (such as support, confidence, leverage, lift).

A typical application of association rule mining is market basket analysis. This process analyses customer-buying habits by finding associations between the different items that customers place in their “shopping baskets”. The discovery of such associations can help retailers develop marketing strategies by finding which items are frequently purchased together by customers.

2.1.3.1 Sequential patterns- Time series Analysis

Sequential pattern mining is the mining of frequently occurring patterns related to time or other sequences. Most studies on sequential pattern mining concentrate on symbolic patterns. The problem of mining sequential patterns can be stated as follows:

Given a potentially large pattern (string) S , we are interested in sequential patterns of the form $a \rightarrow b$, where a , b are substrings inside S , such that the frequency of ab is not less than some minimum support and the probability that a is immediately followed by b is not less than minimum confidence.

In daily and scientific life sequential data are available and used everywhere. Some representative examples are text, music notes, weather data, satellite data streams, business transactions, telecommunications records, experimental runs, DNA sequences, histories of medical records. Discovering sequential patterns can benefit the user or scientist by predicting coming activities, interpreting recurring phenomena or extracting similarities.

2.2. Patterns in Signal Processing - Content-based Music Retrieval

Large-scale storage of sound and music has become possible in the last decade only. In addition, the new possibility for wide-area distribution of multimedia over the Internet has given rise to new requirements for flexible and powerful databases for musical and audio data. One of these requirements is the complexity of a selection query upon a database that contains massive amounts of musical data [PAC00]. For example consider the following question: "I want to browse through Bach Fugues recorded in C minor and performed with a clavichord". It is clear that this kind of queries address information hidden in the content of the music signal and raise the following challenges related to content-based music retrieval: instrument recognition, melody spotting, musical key extraction, musical pattern recognition, composer recognition, music structure extraction and music segmentation, to name but a few.

2.2.1 A survey of existing research efforts

Feature selection is a topic of ongoing research. It turns out that for a system to be able to support several kinds of queries, it should be able to extract a wide range of different kinds of features from the data loaded in the database.

Various researchers have also tried to achieve automatic extraction of the structure of music recordings in an attempt to prove the assumption that music similarity may be considered in part, as a comparison of musical structure. Music is, indeed, often described (at a high level of abstraction) in terms of the structure of repeated patterns. Discovering musical structure in audio recordings has been addressed by [DAN02], [CON02], [MER01], [MAR01], etc.

2.3. Patterns in Information Retrieval

Another research field where patterns are apparent is that of Information Retrieval. In a retrieval setting we have a collection of discourse material, also called *corpus*, and users submit queries to the system in order to retrieve information that suits their interest. Queries are often vaguely defined, in contrast to traditional database systems, due to lack of a query language or algebra. A query consisting of only a few words does not always reflect the user's actual interest; therefore users often experience frustration from a retrieval system. The *Latent Semantic Indexing* (LSI) [DDF+90, BDO95], a retrieval model, unveiling patterns in terms' usage, seems to produce more effective information retrieval.

2.4. Patterns in Mathematics

Mathematics is the science of patterns. Not only do patterns take many forms over the range of school mathematics, they are also a unifying theme. Number patterns, such as 3, 6, 9, 12, e.t.c. are familiar to us since they are among the patterns we first learn as young students. As we advance, we experience number patterns again through the huge concept of functions in mathematics. But patterns are much broader. They can be sequential [AS95], spatial [ST99], temporal [DLMRS98][SB98], and even linguistic [FFKLLRSZ98][LAS97].

The various mathematical patterns can be summarized to the following categories:

2.4.1 Number Patterns

In mathematics we usually come upon numbers that share certain properties. In that case the pattern is the rule or constraint that several items satisfy. Sometimes it is very useful to collect number patterns so as to be able to learn the behaviour of numbers collected by telecommunication data for example [BCH00] or by equation solving [SB99].

2.4.2 Patterns in graphs

Almost every graph belongs to a graph pattern, which means that each graph has some attributes that forces it to obey to a specific behaviour, as one or more of it's variables grow or lessen in value. For example, we can easily distinguish some graph patterns, like linear ($x + y = 7$, $2x + 3 = y$, etc.) and non-linear graphs (parabola, circle [Sch97], ellipse)

For example, Figure 1 depicts the non-linear graphs that present a pattern of their own.

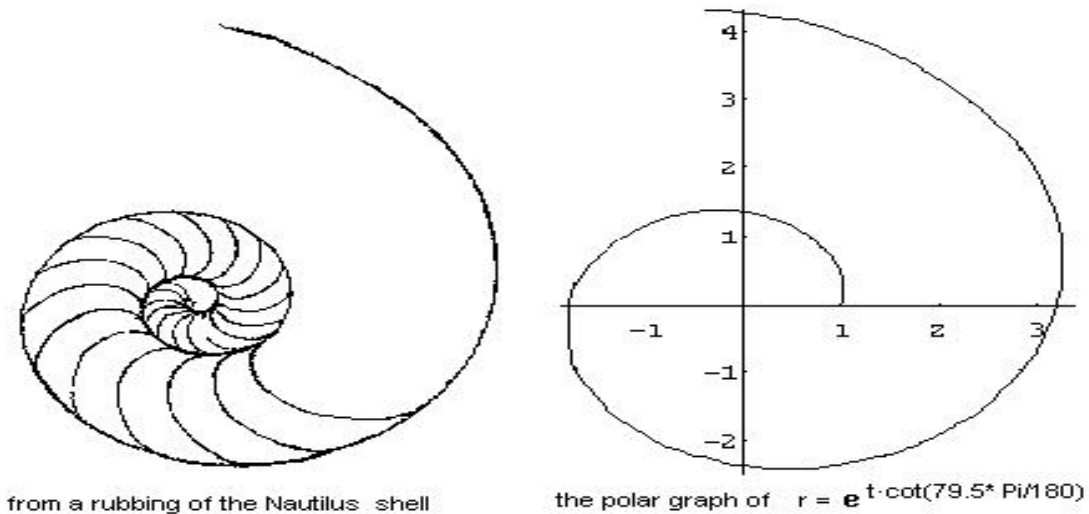


Figure 1. Non-linear graphs

2.4.3 Patterns in shapes

We can also find patterns in shapes. The concept in this case is that the majority of shapes follow a similarity, for example in the number of vertices from which the shape is constituted. So, patterns could be triangles (which might be similar), squares, polygons with n -angles, etc. The existence of such patterns can help us in recognizing familiar shapes in image processing [NPP00] or even for reconstructing polygonal images [CN95].

2.4.4 Patterns in algebra

Algebra provides verbal, symbolic and graphical formats for discussing and representing settings as diverse as the pricing patterns of merchandise in a store, the behavior of a car as it accelerates or slows down, the changes in two chemicals as they react with one another, or the type of variation existing in a comparison of two factors in the economy. In algebra, we could think of many patterns, like the way equations are solved, iterating functions, etc. These patterns apply to fields like music [LK94], moving objects applications [G99], etc.

2.4.5 Patterns in Cryptography

Cryptography is one of the main subjects in which mathematics rays supreme. In cryptography, every cryptographic system could be considered as a pattern itself. For example, if we have many sets of raw data and in each set we enforce a specific encryption, then these sets would share the similarity of their encryption attribute. We can find many such cases in cryptography, like *Vigenere Cipher*, *Caesar Cipher*, *Gronsfeld cipher* etc. Such

Cryptographic patterns are widely used in word processors, electronic commerce systems, spreadsheets, databases and security systems [BRD99].

2.5. Patterns in Information Visualization

Information visualization and visual data mining techniques can help to deal with the large amount of data that is stored in scientific, engineering, and environmental databases [Kei01a]. Visualization techniques are useful for showing an overview of the raw data and detecting patterns [Kei 00]. Patterns are groups of data points in the visualization that represent potentially valuable information and provide new insights for the user. The user needs to be able to zoom and filter the data in order to focus on one and more patterns [AW95].

Visual data exploration can be seen as a hypothesis generation process. The visualization allows the user to identify patterns of interest or groups of related data points and gain insight into the raw data. Visualization can also be used to analyze the patterns on different levels of abstraction, which may result in adapting existing hypotheses or generating new hypotheses.

3. Related Work

In this section we address the current and evolving efforts on modeling data mining processes such as the work of the Data Mining Group and the specification of the Predictive Model Markup Language, the SQL/MM standard, the Common Warehouse Model, the Java Data Mining API and PQL, a pattern query language developed by Information Discovery, Inc.

3.1. Data Mining Group / Predictive Model Markup Language [DMG]

The Data Mining Group (DMG) is an independent, vendor led group¹ that develops data mining standards, such as the Predictive Model Markup Language (PMML). PMML is a collection of XML Document Type Descriptors (DTDs) that provide a uniform way for modeling data mining processes and results. In the next sections we will present the main features of the PMML DTDs.

¹ In January 2003, the members of DMG were: *Angoss Software Corp. Toronto, CAN, IBM Corp. Somers NY, NCR Corp. Dayton OH, Magnify Inc. Chicago IL, Oracle Corporation Redwood Shore CA, National Center for Data Mining University of Illinois at Chicago, SPSS Inc. Chicago IL, Xchange Inc. Boston MA, MINEit Software Ltd. Bracknell UK.*

PMML defines a variety of specific mining models such as tree classification, neural networks, regression, etc.

3.1.1 General Structure of a PMML Document

PMML uses XML to represent mining models. The structure of the models is described by a DTD, which is called the PMML DTD. The DTD that all PMML documents must conform is illustrated in Figure 2(a).

3.1.2 Header

The Header DTD is illustrated in Figure 2(b).

- *Header*: The top-level tag that marks the beginning of the header information.
- *Copyright*: This head attribute contains the copyright information for this model.
- *Description*: This head attribute contains a non-specific description for the model. This attribute should only contain human readable information, and models mentioned in this dtd file should not be expected to utilize the information contained in this attribute.
- *Application*: This head element describes the software application that generated the PMML.
- *Name*: The name of the application that generated the model.
- *Version*: The version of the application that generated this model.
- *Annotation*: Document modification history is embedded here.
- *Timestamp*: This element allows a model creation timestamp in the format YYYY-MM-DD hh:mm:ss GMT +/- xx:xx.

3.1.3 Settings

The element Settings can contain any XML value describing the configuration of the training run that produced the model instance. This information is not directly needed in a PMML consumer, but in many cases it is helpful for maintenance and for visualization of the model. The content of Settings is not defined in PMML 2.0.

3.1.4 Data Dictionary

The data dictionary contains definitions for fields as used in mining models. It specifies the types and value ranges. These definitions are assumed to be independent of specific data sets as used for training or building a specific model.

A sample data dictionary is illustrated in Figure 2(c). “numberOfFields” is the number of fields, which are defined in the content of DataDictionary, and this number can be added for consistency checks. “DataField” must be unique in the data dictionary. “displayName” is a string, which may be used by applications to refer to that field. Within the XML document only the value of name is significant. If “displayName” is not given, then “name” is the default value. The “optype” attribute defines the operations that apply on fields’ values.

3.1.5 Transformation Dictionary (Derived Values)

At various places the mining models use simple functions in order to map user data to values that are easier to use in the specific model. For example, neural networks internally work with numbers, usually in the range [0..1]. Numeric input data are mapped to the range [0..1], and categorical fields are mapped to series of 0/1 indicators. Similarly, Naive Bayes models internally map all input data to categorical values. For PMML to be able to handle such mappings it defines various kinds of simple data transformations:

- Normalization: map values to numbers, input can be continuous or discrete.
- Discretization: map continuous values to discrete values.
- Value mapping: map discrete values to discrete values. Mapping missing values as a special case of value mapping.
- Aggregation: summarize or collect groups of values, e.g. compute average.

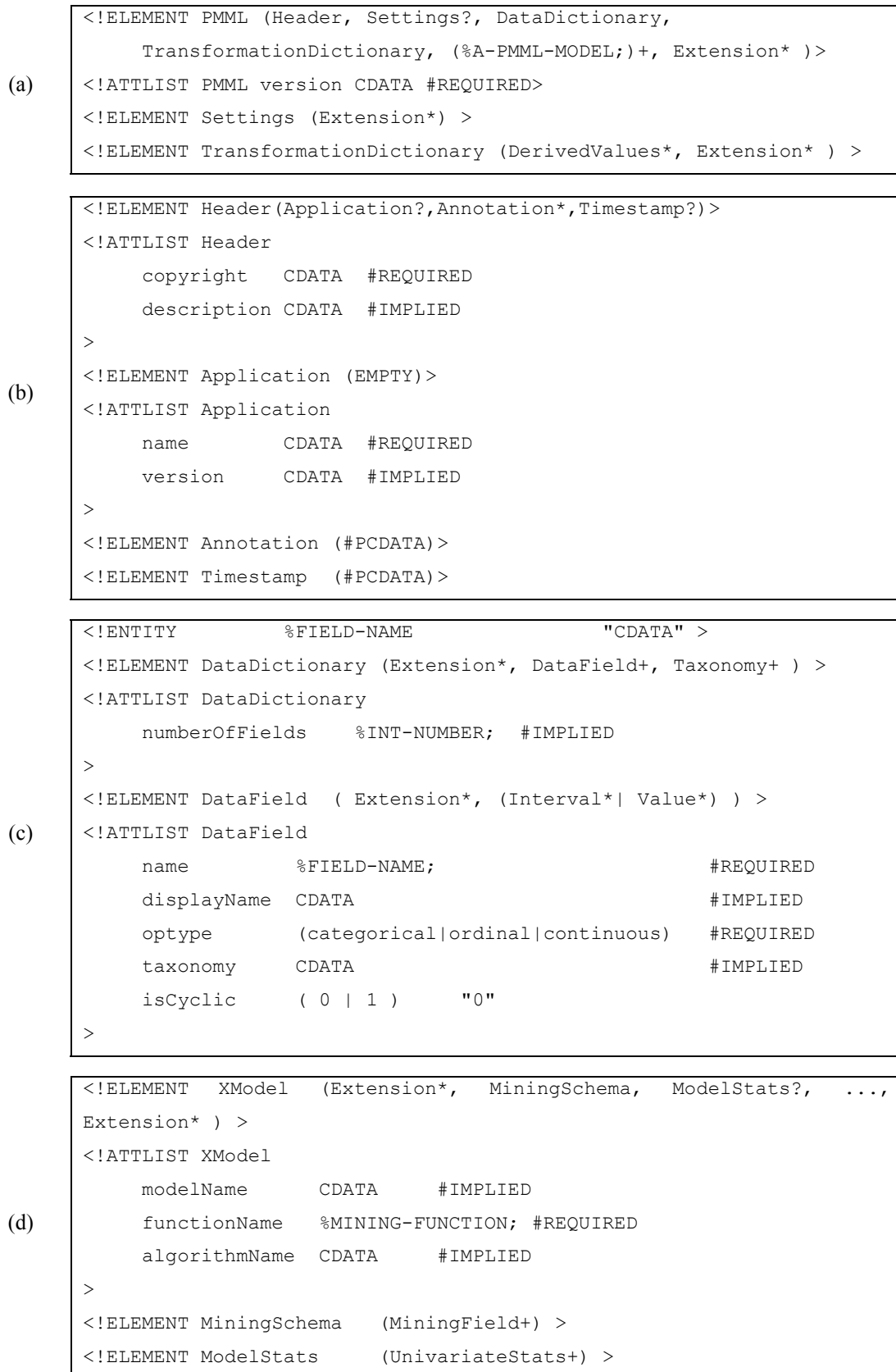


Figure 2. (a) DTD, (b) header and (c) data dictionary of a PMML document

3.1.6 PMML mining models

A PMML document can contain more than one model. PMML supports the following data mining models:

- *TreeModel*: The tree modeling framework allows for defining either a classification or prediction structure. Each Node holds a rule, called PREDICATE, that determines the reason for choosing the Node or any of the branching Nodes.
- *NeuralNetwork*: A neural network has one or more input nodes and one or more neurons. Some neuron's outputs are the output of the network. The network is defined by the neurons, their connections and the corresponding weights. All neurons are organized into layers; the sequence of layers defines the order in which the activations are computed.
- *ClusteringModel*: PMML models for Clustering are defined by two different classes. These are *center-based* and *distribution-based* cluster models. In center-based models a cluster is defined by a vector of center coordinates. Some distance measure is used to determine the nearest center, that is the nearest cluster for a given input record. For distribution-based models (e.g. in demographic clustering) the clusters are defined by their statistics. Some similarity measure is used to determine the best matching cluster for a given record.
- *RegressionModel*: The regression functions are used to determine the relationship between the dependent variable (target field) and one or more independent variables. The dependent variable is the one whose values you want to predict, whereas the independent variables are the variables that you base your prediction on. PMML defines three types of regression models: linear, polynomial, and logistic regression.
- *NaiveBayesModels*: Naive Bayes uses Bayes' Theorem, combined with a ("naive") presumption of conditional independence, to predict, for each record (a set of values, one for each field), the value of a target (output) independence, to predict, for each record (a set of values, one for each field), the value of a target (output) field, from evidence given by one or more predictor (input) fields.
- *AssociationModel*: The Association Rule model represents rules where some set of items is associated to another set of items. For example a rule can express that a certain product is often bought in combination with a certain set of other products.

-
- *SequenceMiningModel*: The basic data model consists of an Object, identified by the “Primary Key” that has a number of events attributed to it, defined by the “Secondary Key”. Each event consists of a set of ordered items. An “Order Field” defines the order of the items within an event, with an optional qualifier in the form of an attribute name.

For every model there is a corresponding DTD that describes the metadata and the processes of each model. In the following we give a description for the generic framework of PMML that is used by every data model. For all PMML models the structure of the top-level model element is similar to the one illustrated in Figure 2(d).

The naming conventions for PMML are “ElementNames” in mixed case, first uppercase “attributeNames” in mixed case, first lowercase “enumConstants” in mixed case, first lowercase “ENTITY-NAMES” all uppercase. The character '-' is used less frequent in order to avoid confusion with mathematical notation.

3.2. SQL Multimedia and Application Packages (SQL/MM)

SQL for Multimedia and Application Packages (SQL/MM) [SQL/MM] is a standard based on SQL that has been developed by the International Organization for Standardization (ISO). SQL/MM Data Mining defines SQL structured user-defined types (including methods over them) to address data mining tasks. These structured types are first-class SQL types that can be accessed through SQL:1999 base syntax. These accesses also include invocation of the routines (methods) associated with the structured types.

The standard supports the following models: Rule model, Clustering model, Regression model and Classification model. Every model has a corresponding SQL structured user-defined type. A set of predefined types completes the full definition of each model. The basic type is named DM_*Model where “*” is replaced by ‘Class’, ‘Rule’, ‘Clustering’, ‘Regression’ for a classification, rule, clustering and regression model respectively.

- DM_*Settings type: Instances of that type are used for storing various parameters of the data mining model, such as the maximum number of clusters or the depth of a decision tree.
- DM_*TestResult: Instantiations of that type hold the results of the testings during the training phase of the data mining models.
- DM_*Result: The running of a data mining model against real data creates instances of that type.

- DM_*Task: Instances of that type store metadata that describe the process and control of testings and of the actual runnings.

Example (Rule Model): The DM_RuleModel type represents models, which are the result of the search for association rules. In Figure 3, we give the definition of the Rule Model type as described in the standard.

```
CREATE TYPE DM_RuleModel
AS (
  DM_content CHARACTER LARGE OBJECT(DM_MaxContentLength)
)
INSTANTIABLE
NOT FINAL
STATIC METHOD DM_impRuleModel
(input CHARACTER LARGE OBJECT(DM_MaxContentLength))
RETURNS DM_RuleModel
LANGUAGE SQL
DETERMINISTIC
CONTAINS SQL
RETURNS NULL ON NULL INPUT,
```

Figure 3. An example of association rule model in SQL/MM

The Rule Model type has only one member variable, *DM_content*. In that variable, that is a CHARACTER LARGE OBJECT (CLOB), the complete information about one instance of the model is stored. Method *DM_impRuleModel* takes a CLOB as input parameter and if it is a proper representation of a *DM_RuleModel*, then a new value of type *DM_RuleModel* is created. A CLOB is a proper representation of a *DM_RuleModel*, if it is a valid instance of the PMML Association Rules DTD according to XML.

3.3. Common Warehouse Model (CWM)

3.3.1 Overview

The main purpose of CWM [CWM] is to enable easy interchange of warehouse and business intelligence metadata between warehouse tools, warehouse platforms and warehouse metadata repositories in distributed heterogeneous environments. CWM is based on three key industry standards:

- UML - Unified Modeling Language, an OMG modeling standard

- MOF - Meta Object Facility, an OMG metamodeling and metadata repository standard
- XMI - XML Metadata Interchange, an OMG metadata interchange standard

The CWM provides a framework for representing metadata about data sources, data targets, transformations and analysis, and the processes and operations that create and manage warehouse data and provide lineage information about its use. The CWM Metamodel consists of a number of sub-metamodels which represent common warehouse metadata in the following major areas of interest to data warehousing and business intelligence. In this section we give an overview of the Data Mining sub-metamodel.

3.3.2 Data Mining Metamodel

The CWM Data Mining metamodel represents three conceptual areas: (i) the overall Model description, (ii) Settings and (iii) Attributes.

The Model conceptual area consists of a generic representation of a data mining model (that is, a mathematical model produced or generated by the execution of a data mining algorithm). This consists of MiningModel, a representation of the mining model itself, MiningSettings, which drive the construction of the model, ApplicationInputSpecification, which specifies the set of input attributes for the model, and MiningModelResult, which represents the result set produced by the testing or application of a generated model.

3.4. Java DM API

Java DM API (JDMAPI) [JDM] follows SUN's Java Community Process as a Java Specification Request (JSR). It addresses the need for an API that will give "procedural" support to all the existing and evolving data mining standards².

3.5. Oracle9i Data Mining

Oracle has embedded data mining within the Oracle9i database with Oracle9i Data Mining (ODM) [Oracle9i]. All the functionality for Oracle9i data mining operations, such as model

² The group that participates in the specification of the API is constituted by the following members: *BEA Systems, Blue Martini Software, Dubitzky, Werner, Hyperion Solutions Corporation, IBM, Kana Communications Inc., Magnify Research, Inc., MINEit Software Limited, Oracle, Quadstone, SAP AG, SAS Institute, SPSS, Strategic Analytics and Sun Microsystems, Inc.*

building, scoring functions, and testing is provided via a Java API. Oracle9i Data Mining consists of the following components:

- Oracle9i Data Mining (ODM) API
- Data Mining Server (DMS)

The ODM API allows users to write programs that perform data mining operations. It is based on the proposed concepts of the Java Data Mining API (cf. Section 3.4). Oracle9i Data Mining supports two data mining functions: classification for supervised learning and association rules for unsupervised learning. The mining functions use two algorithms: Naive Bayes and Association Rules.

The most important object, in context of storing data mining results, is the mining result object. A mining result object contains the end products of test or build-model mining operations.

Oracle intends to support other models in future versions, such as Decision Trees and Classification models. It also intends to provide full support of SQL/MM and PMML.

3.6. Information Discovery DataMining Suite

Information Discovery [IDMS] has developed a set of tools and systems for data mining and knowledge extraction. These products make use of the Pattern Query Language (PQL). PQL is a pattern-oriented query language specifically designed to provide business users access to refined information. No other information is available (at least through their web site) about the syntax or the semantics of the language. Its notion is very close to the motive of the PANDA project and that is the reason for mentioning it in this survey.

4. Conclusion

The database research community claims that technology has reached a critical point since there are certain requirements and information types that are either partially supported or not supported at all [B+98]. The innovation of the PANDA project lies in the vision for a new approach aiming at the definition of a system architecture that efficiently represents, maintains and manages patterns. It refers to a variety of domains, that is, among others, knowledge discovery in (traditional or non-traditional) databases, time-involving applications (time series or moving objects databases), multimedia systems (image or video databases), scientific data, and the WWW (as a huge repository of unstructured information). The

cornerstone of this new approach will be the **pattern concept**, aiming at representing huge volumes of information in an effective way.

Moreover, PANDA aims at the integration of existing approaches towards a novel logical integration of patterns into a data model, language and base management system support.

Regarding PMML and the other modelling approaches, the proposed system architecture is vertical. It heads in a vertical approach defining an extensible type system. As it has already been discussed in the previous sections, patterns arise from different scientific fields (i.e., data mining, mathematics, information retrieval etc). PANDA aims at supporting any pattern type regardless to the application, while other approaches are mainly oriented to data mining patterns.

The majority of users both in scientific and business field do not want massive volumes of data, but they are interested in the patterns and trends hidden within data. Since these patterns need to be accessed, manipulated and managed, just as data elements are managed the concept of "pattern management" is introduced. Pattern management systems deal with patterns, like data management systems deal with data. Moreover, they require distinct repositories and query languages, corresponding to languages that have been developed for data management.

In this paper, we reviewed different types of patterns from many areas, the current efforts on modeling data mining operations were addressed along with the corresponding results. Furthermore different procedures of pattern extraction were found which give us a brief idea of the diversity between various patterns types. After a close observation of the various pattern types and given the informal definition that *pattern is a compact and rich in semantics representation of raw data* [R+03], we draw the conclusion that there are many common characteristics between all of them.

One of the challenges in the field of pattern mining (or pattern recognition) is the development of a framework capable of representing and dealing with every kind of patterns independently of the application and/or the method used to extract patterns. This framework has to serve as a precise and conceptual foundation for the representation and behavior of patterns. It will be the basis for the design and development of a system that handles (i.e., stores, processes, and retrieves) patterns and supports pattern-related operations.

To proceed with the definition of the considered framework we have to define the structure and the requirements for representing each kind of patterns while the relationships of

them have to be identified. Also it is important that we identify the behavior of patterns and the functions that a pattern-related system has to support.

References

- [AR94] R. Agrawal, S. Ramakrishnan. "Fast Algorithms for Mining Association Rules". *Proc. of the 20th VLDB Conference*, 1994.
- [AR95] R. Agrawal, S. Ramakrishnan: *Mining Sequential Patterns*, IBM Almaden Research Center, 1995.
- [AT99] A. A. Abel-Samad and A. H. Tewfik: *Search Strategies for radar Target localization*, University of Minnesota, Minneapolis, 1999.
- [AW95] C. Ahlberg and E. Wistrand. IVEE: An information visualization and exploration environment. *Information Visualization*, Atlanta, GA pages 66–73, 1995.
- [B+98] P. A. Bernstein et al., "The Asilomar Report on Database Research", *SIGMOD Record*, 27(4):74-80, December 1998.
- [BCH00] A. Baritchi, D. J. Cook, and L. B. Holder: *Discovering Structural Patterns in Telecommunications Data*, Texas, 2000.
- [BDO95] M. W. Berry, S. T. Dumais, G.W. O'Brien, Using linear algebra for intelligent Information retrieval, *SIAM Review*, 37(4): 573-595, 1995
- [BRD99] A. M. Braga, C. M.F. Rubira and R. Dahab: *Tropyc: A Pattern Language for Cryptographic Software*, Brazil, 1999.
- [CN95] P. Clifford and G. Nichols: *A Metropolis Sampler for Polygonal Image Reconstruction*, UK, 1995.
- [CON02] D. Conklin, "Representation and Discovery of vertical patterns in Music", *Lecture Notes in Artificial Intelligence (LNAI) 2445*, Springer Verlag, 2002.
- [CS96] P. Cheeseman, J. Stutz. "Bayesian Classification (AutoClass): Theory and Results". *Advances in Knowledge Discovery and Data Mining*. (Eds:U. Fayyad,et al), AAAI Press,1996.
- [CWM] Common Warehouse Metamodel (CWM). Available at <http://www.omg.org/cwm>
- [DAN02] R. Dannenberg and N. Hu, "Discovering Musical Structure in Audio Recordings", *Lecture Notes in Artificial Intelligence (LNAI) 2445*, Springer Verlag, 2002.

-
- [DDF+90] S. Deerwester, S. T. Dumais, G. Furnas, Th. K. Landauer, R. Harshman, Indexing by Latent Semantic Analysis, *Journal of the Society for Information Science*, 41(6): 391-407, 1990.
- [DLL+97] S. T. Dumais, T. A. Letsche, M. L. Littman, T. K. Landauer, Automatic cross-language retrieval using Latent Semantic Indexing, *In AAAI Spring Symposium on Cross-Language Text and Speech Retrieval*, March 1997.
- [DLM+98] G. Das, K.-L. Lin, H. Mannila, G. Renganathan, and P. Smith: *Rule discovery from time series*, USA, 1998.
- [DMG] DMG, Predictive Model Markup Language (PMML). Available at http://www.dmg.org/pmmlspecs_v2/pmml_v2_0.html
- [FFK+98] R. Feldman, M. Fresko, Y. Kihar, Y. Lindell, O. Liphstat, M. Rajman, Y. Schler, O. Zamiv: *Text Mining at the Term Level*, 1998.
- [Gav99] D.M. Gavrilla: *The Visual Analysis of Human Movement: A Survey*, Germany, 1999.
- [Gra02] J. Gray. "The Information Avalanche: Reducing Information Overload". *Onassis Foundation Science Lecture Series*, Heraklion, Crete, Greece, 15-19 July 2002. Slides available at <http://research.microsoft.com/~Gray/Talks/>
- [GRS98] S. Guha, R. Rastogi, K. Shim. "CURE: An Efficient Clustering Algorithm for Large Databases", *Proceedings of ACM SIGMOD Conference*, 1998.
- [HK01] J. Han, M. Kamber. *Data Mining: Concepts and Techniques*. Academic Press, 2001.
- [IDMS] Information Discovery DataMining Suite. Available at <http://www.patternwarehouse.com/dmsuite.htm>
- [ISO] ISO SQL/MM Part 6. Available at http://www.sql-99.org/SC32/WG4/Progression_Documents/FCD/fcd-datamining-2001-05.pdf
- [JDM] Java Data Mining API. Available at <http://www.jcp.org/jsr/detail/73.prt>
- [Kei00] D. A. Keim: "Designing Pixel-oriented Visualization Techniques: Theory and Applications", *Transactions of Visualization and Computer Graphics*, 2000.
- [LAS97] B. Lent, R. Agrawal, and R. Srikant: Discovering Trends in Text Databases. In *Proceedings of the 3rd International Conference on Knowledge Discovery (KDD)*, (1997).
- [LK94] E. W. Large and J. F. Kolen: *Resonance and the Perception of Musical Meter*, The Ohio State University, 1994.
- [LV00] P. Lyman and H.R. Varian, "How Much Information", 2000. Available at <http://www.sims.berkeley.edu/how-much-info>

- [MAR01] A. Marsden, "Representing melodic patterns as networks of elaborations", *Computers and the Humanities*, 35:37-54, 2001.
- [MER01] D. Meredith, G. Wiggins and K. Lemstrom, "Patterns induction and Matching in Music and other multidimensional datasets", *Proceedings of the Conference on Systemics, Cybernetics and Informatics*, volume X, 2001.
- [Mit97] T. Mitchell. *Machine Learning*. McGraw-Hill, 1997
- [NPP00] Chichahito Nakajima, Massimiliano Pontil and Tomaso Poggio: *People Recognition and Pose Estimation in Image Sequences*, JAPAN, 2000.
- [Ora] Oracle9i Data Mining Concepts. Available at http://otn.oracle.com/docs/products/oracle9i/doc_library/release2/datamine.920/a95961/1concept.htm#923516
- [PANDA] PANDA "Patterns for Next-generation Database Systems", IST-2001-33058. Project site at <http://dke.cti.gr/panda>
- [PRC00] F. Pachet, P. Roy, D. Cazaly, "A Combinatorial approach to content-based music selection", *IEEE Multimedia*, Vol. 1, 2000.
- [R+03] S. Rizzi et al., "Towards a Logical Model for Patterns". *Proceedings of the 22nd Int'l Conference on Conceptual Modeling (ER'03)*, Chicago, IL, 2003.
- [RA95] S. Ramakrishnan, R. Agrawal. "Mining Generalized Association Rules". *Proc. of the 21st VLDB Conference*, 1995.
- [RS98] R. Rastori, K. Shim. "PUBLIC: A Decision Tree Classifier that Integrates Building and Pruning". *Proceedings of the 24th VLDB Conference*, New York, USA, 1998.
- [SB98] N. Sumpter and A. J. Bulpitt: *Learning Spatio-Temporal Patterns for predicting Object behaviour*, UK, 1998.
- [SB99] S. Schulz and F. Brandt: *Using Term Space Maps to Capture Search Control Knowledge in Equational Theorem Proving*, Germany, 1999.
- [Sch97] O. Schram: *Circle Patterns with the combinatorics of the square grid*, The Weizmann Institute, 1997.
- [TK98] S. Theodoridis, K. Koutroumbas: *Pattern Recognition*, Academic Press, 1998.

Table of Contents

| | | |
|-------|---|----|
| 1. | Introduction - Motivation | 3 |
| 2. | Application fields and pattern usage | 4 |
| 2.1. | Patterns in Data Mining | 4 |
| 2.1.1 | Clustering..... | 5 |
| 2.1.2 | Classification – Decision Making..... | 6 |
| 2.1.3 | Association Rules | 7 |
| 2.2. | Patterns in Signal Processing - Content-based Music Retrieval | 8 |
| 2.2.1 | A survey of existing research efforts | 8 |
| 2.3. | Patterns in Information Retrieval | 9 |
| 2.4. | Patterns in Mathematics | 9 |
| 2.4.1 | Number Patterns | 9 |
| 2.4.2 | Patterns in graphs..... | 9 |
| 2.4.3 | Patterns in shapes..... | 10 |
| 2.4.4 | Patterns in algebra..... | 10 |
| 2.4.5 | Patterns in Cryptography | 10 |
| 2.5. | Patterns in Information Visualization | 11 |
| 3. | Related Work..... | 11 |
| 3.1. | Data Mining Group / Predictive Model Markup Language [DMG] | 11 |
| 3.1.1 | General Structure of a PMML Document..... | 12 |
| 3.1.2 | Header..... | 12 |
| 3.1.3 | Settings | 12 |
| 3.1.4 | Data Dictionary..... | 13 |
| 3.1.5 | Transformation Dictionary (Derived Values)..... | 13 |
| 3.1.6 | PMML mining models..... | 15 |
| 3.2. | SQL Multimedia and Application Packages (SQL/MM)..... | 16 |
| 3.3. | Common Warehouse Model (CWM)..... | 17 |
| 3.3.1 | Overview | 17 |
| 3.3.2 | Data Mining Metamodel..... | 18 |
| 3.4. | Java DM API..... | 18 |
| 3.5. | Oracle9i Data Mining..... | 18 |
| 3.6. | Information Discovery DataMining Suite..... | 19 |
| 4. | Conclusion..... | 19 |
| | References | 21 |