Intelligent Data Analysis: Issues and Challenges

Richi Nayak
School of Information Systems
Queensland University of Technology
Brisbane, QLD 4001, Australia
r.nayak@qut.edu.au

ABSTRACT

Today with the advances of technology, mountainous amounts of data are now available in science, business, industry and many other areas. Evaluation of these collected data may lead to the discovery of trends and patterns hidden within the data that increase the working efficiency and improve the quality of decision making. Of course this advantage comes with a price. It is becoming more and more difficult to gain some valuable information when analysing with these increasing data sets. This paper attempts to discuss a wide range of problems that may appear while analysing the data, and suggests strategies to deal with them. Some of these problems and suggestions are examined with the results of data analysis on a real-life example of risk assessment of level crossing data.

Keywords: data analysis, data mining, risk assessment of level crossing, rule extraction, neural networks, rule induction

1. INTRODUCTION

As more and more data being collected and stored everyday, various data analysis techniques have been developed based on the works of pattern recognition, statistic, artificial intelligence, machine learning, database system, internet/intranet and others. An intelligent data analysis (IDA) task includes knowledge discovery, prediction, process/system modelling or building knowledge based systems. There are many achievements of applying IDA methods in various areas such as marketing, medical, financial and agriculture. IDA tools and applications have generated positive results and continuously stimulated exploring new application areas due to the benefits brought by this technology.

The rapidly expanding volume of real-time data, resulting from the explosion in activity from the web, multimedia, electronic commerce and others, has contributed to the demand for and provision of more sophisticated IDA methods [13]. The general idea of analysing the large amounts of data with rich description is both appealing and intuitive, but technically it is significantly challenging and difficult. There must be some strategies that should be implemented for better use of data collected from such large and complex data sources.

This paper addresses a wide range of problems that may appear during a data analysis process, suggests some strategies to handle them, and identifies challenging areas for further research. Before discussing technical problems in intelligent data analysis and their ramifications, we briefly introduce a typical data analysis process and various possible tasks and techniques. This paper also presents a case study of risk assessment of the Queensland Rail level-crossing dataset. An integrated data analysis system utilising machine learning techniques is used to analyse this industry application. The empirical results on the level-crossing database demonstrate that machine learning techniques can be applied to discover hidden information from real-life datasets with high accuracy and good comprehensibility.

2. AN EXAMPLE IDA PROCESS

Intelligent data analysis is a process of finding useful and interesting structures from the data, thus assisting in decision making [8]. A typical IDA process starts with identifying a problem depending on the interest of a data analyst. Next, all sources of information are identified and a subset of data is generated from the accumulated data for the IDA application. To ensure quality, the data set is pre-processed by removing noise, handling missing information and transforming to an appropriate format. An IDA technique or a combination of techniques appropriate for the type of knowledge to be discovered is then applied to the derived data set. The discovered knowledge is then manipulatated, evaluated and interpreted, typically involving some post-processing tools such as visualization techniques. Finally the information is presented to user. Sometimes this process includes the maintenance of results by iterating all the steps again for user satisfaction, or/and adapting the new information in the future.

Usually the gained knowledge is a type of classification rules, characteristic rules, association rules, functional relationships, functional dependencies, causal rules, temporal knowledge and/or clusters.

3. VARIOUS DATA ANALYSIS TASKS AND TECHNIQUES

According to the goals and interests of an end user, such as characterising the contents of data set as a whole or establishing links between subsets of patterns in the data set, a data analysis process can have three possible tasks - predictive modelling, clustering and link analysis [3].

The goal of predictive modelling is to make predictions based on essential characteristics about the data. The goal is to build a model to map a data item into one of the several predefined classes or to a real-valued prediction variable. Any supervised machine learning algorithm, that learns a model on previous or existing data, can be used to perform predictive modelling. The model is given some already known facts with correct answers, from which the model learns to make accurate predictions. Neural networks, decision trees, bayesian classifiers, K-nearest neighbour classifiers, case based reasoning, genetic algorithms, rough set and fuzzy set are some of the approaches used for mapping discrete-valued target variables. Regression techniques, induction trees, neural networks and radial basis function are some of the approaches used for mapping continuous-valued target variables.

The goal of clustering is to identify items with similar characteristics, and thus creating a hierarchy of classes from the existing set of events. Any unsupervised machine learning algorithm, for which a predetermined set of data categories is not known for the input data set, can be used to perform clustering. The model is given some already known facts, from which the model derives categories of data with similar characteristics. Some major clustering methods are partitioning, hierarchical, density based and model based algorithms [7].

The link analysis establishes internal relationship among items in a given data set. This goal is achieved by association discovery, sequential pattern discovery and similar time sequence discovery tasks [3]. These tasks expose samples and trends by predicting correlation of items that are otherwise not obvious. The link analysis techniques are based on counting occurrences of all possible combination of items. Some of the most widely used algorithms are Apriori and its variation [2].

4. A CASE STUDY: RISK ASSESSMENT ON LEVEL-CROSSING

The objective of this case study is to assist QR (Queensland Rail) personnel to improve safety measures to avoid level-crossing accidents by identifying accident-prone cases. The set of objects consisting of labelled data patterns comprises the training set for data analysis. The label identifies the group as *Risky* or *Safe* depending on whether or not an accident has happened at the corresponding crossing to which the object belongs. Other attributes specify the property (characteristics of level-crossings in Queensland) of the object. The goals are to identify (1) the features, either individually or in groups, that are responsible for the risk of a level crossing, and (2) the future (unseen) cases as *Risky* or *Safe*. As a result, the types of protections required to achieve an acceptable level of risk will be identified.

The Data Analysis System

Machine learning techniques are one of the most widely used methods for data analysis because of their capability to identify distinguished patterns relating to the concepts to be taught [11]. We have included two forms of machine learning to perform predictive modelling: neural networks and symbolic rule induction. Both neural networks and rule induction techniques have a proven track record in many data analysis and decision-support applications [8, 11, 12, 15]. The two different learning techniques are used because they have different strengths and weakness, and one can outperform another for a certain type of tasks. For example, neural networks perform better for *parallel tasks* (where all the input variables are relevant to the classification) and rule induction methods are more suitable for *sequential tasks* (where the relevance of a particular input variable depends on the values of other input variables) [12,15].

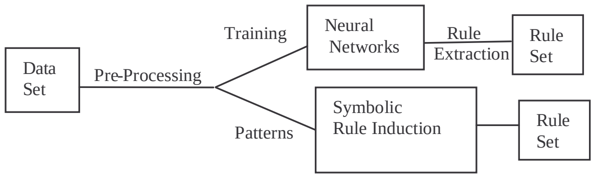

Figure 1: The data analysis system

Based on Neural Networks Neural networks are a powerful general purpose tool applied to prediction and clustering tasks [11]. The ability to learn and generalize from the data that mimics human capability of learning from experiences, makes neural networks useful for data analysis tasks. However there are two fundamental considerations for not using neural networks commonly, (1) the poor comprehensibility of the learned models (an absence of the ability to explain the decision process in a comprehensive form) and (2) the lack of the ability to inducing models from large data sets (requires a long time to train).

We utilize the GYAN methodology [12] to overcome these problems (1) by reforming the connection weights representing the network into a symbolic description known as *rule extraction* (2) by applying features selection before training of the network on the data and then pruning of the trained network. Basic principle of the pruning algorithm is first group the network's links of similar weights in clusters, and then eliminate the clusters whose magnitude is sufficiently low ($|\sum \text{weights}| < \text{Bias}$) such that they are not contributing towards the network's decision.

The neural networks are trained by the cascade correlation algorithm [5] to avoid guessing the number of hidden nodes. The algorithm starts by generating initial network topologies based on the structure and types of the data, and then dynamically modifying topologies by training the output nodes(s) to approximate the target function. Networks are trained until their performance ceases to improve.

Rule extraction techniques are applied to interpret the knowledge embedded in pruned (optional) and trained networks. The final output is in the form of propositional rules or constrained first order rules (recursive functions are not allowed) depending upon the problem and the user request. If the number of the extracted propositional rules are very large or/and hard to understand, the rules may be generalized into constrained first order rules by utilizing the generalization algorithm based on the Plotkin's least general generalization concept [12].

We have adopted two types of rule extraction techniques in the GYAN methodology. The reason of using two different types of rule extraction techniques is that each one of them treats rule extraction in very different manner. A *pedagogical* rule extraction approach treats the trained ANN as a *'black box'*.

The rule extraction task is viewed as a *learning task* where the target concept is the function computed by the trained network and the input features are simply the network's input features. The objective is to extract a set of rules that characterizes the target concepts directly in terms of the inputs. *Decompositional rule extraction* approaches whereas extract rules by decomposing a multi layer network into a collection of single layer networks or nodes. The aim is to extract rules at the level of each individual hidden and output node, and then aggregate to form the composite rule base that describes the network as a whole. The rationale is that the function learnt by the trained network is easier to express in terms of intermediate concepts and in turn, the intermediate concepts are easier to express in terms of original attributes. We use the *RuleVI* pedagogical and the *LAP* decompositional rule extraction algorithms [14].

Based on Rule-Induction Techniques The rule-induction algorithms that we use for data analysis are C5 [16], one of the current best decision-tree based classifier and FOIL [17], one of the current best first-order inductive learners. The C5 algorithm constructs a decision tree from a set of examples by selecting the most informative attribute according to a gain criterion at each step. The algorithm proceeds by selecting a subset of the training examples to construct a decision tree. Attributes are evaluated at each step (level) to form descendant nodes. The attribute selection is based on a 'statistical test' known as the *information gain ratio* criterion to determine how well a given attribute (alone) separates the training examples according to their target classification. If the generated tree does not give the satisfactory classification (acceptable classification accuracy) for all the objects, selections of the rest of the training examples are added, and the process continues until the correct decision set is found. It also includes pruning of the constructed tree to improve generality of the extracted rule set.

Sometimes rule induction techniques, using propositional logic for the data and hypothesis description language, are not sufficient for data analysis where the data model is usually described with several relational tables. In such cases, inductive logic programming (ILP) systems using the first-order logic representation formalism perform better [17]. ILP algorithms sometimes perform worse on learning tasks where propositional description suffices. Also learning within the first-order framework is much harder than in the propositional framework because of the infinitely large space to search for and a more complex and underlying inference mechanism [12]. FOIL first-order rule induction learner has proved to be an effective and efficient method on several learning tasks [17]. FOIL handles the problem of scalability to the large size data by imposing a few restrictions on the hypothesis and the output language such as only learning definite Horn clauses, etc.

Data Pre-processing

The original QR dataset contains two relational tables: accident history and level crossings. The level crossing table contains all the important information about level-crossings, surrounding situations and the vehicle itself. The accident history table contains information about all the accidents that have happened at level crossings in Queensland, Australia. The accident history data table was only utilized to categorize the level-crossing instances as *Risky* or *Safe* (based on the matching of the unique *Level crossing ID*) according to the constituent value {accident or not} in one of its fields, *consequence type*. The non-meaningful attributes, such as *Level crossing ID, FMS branch name, LS code, Nearest Station, Km, Road Name, Source, and Comments*, were excluded for analysis as they were only stored for identification purposes and were not expected to contribute towards the final outcome.

A large portion of desirable data is missing and most is impossible to retrieve as they are collected from operational sources. We address the missing values in three dimensions: patterns, attributes, and values within an attribute. Our decision is based on the quantity of missing data and whether the data is representing the *Safe* case.

Patterns: If a large quantity of data is missing from a pattern (only 30% of attributes were filled with the values), the pattern was ignored if it was representing the *Safe* case. Since, there is already a very large portion of data belongs to *Safe* cases.

Attributes: There were some attributes that had very low distribution in comparison to other attributes in the data because of the missing values. For example the attribute *Pedestrian protection* had values missing in 99.8% of the cases. This type of attribute is not included in the analysis if there is no significant numbers of patterns containing this attribute falling into the category of *Risky*.

Values: A deliberate decision had to be made whether to overlook a missing value or to delete it or to replace it within an attribute. Firstly, the relative distribution of values in the attributes is determined to handle missing values. If there was a large percentage of missing values for an attribute (more than 30%), the lack of information was treated as a valuable indication, and was considered as a special value to be included additionally in the attribute domain. If a small quantity of data is missing from an attribute (say, less than 15%), the missing values were simply disregarded, and during data-transformation these values were given consideration. For example, all the bits in the sparse-coded representation were set to $1/N$ for the missing input value of an attribute with N possible values, rather than applying normal coding in which only one bit has a value at a time.

Another pre-processing area to look at is corrupt data. The level crossing dataset contains noise because of the different sources used in collection. Most of the discrepancies in data were caused by using differing coding schemes. For example, in some cases attributes such as the *Pedestrian density* and *School Children* had values entered in *{high, medium, low} {yes, no}* and in some places in *{yes, no}*. In fact, the *'yes'* values should be one of *'high'*, *'medium'* or *'low'* or vice versa. Such logically impossible or inapplicable values were replaced by the correct values (more generally by the most frequent value).

As a result, several changes were made to the original level-crossing data and finally the duplicated instances were removed.

Data Analysis

The modified data after pre-processing has been presented to each data analyser - GYAN, FOIL and C5. For each method, 10-fold cross-validation tests were performed and the results for the best classifiers are reported. There was very little deviation in the results of all 10 classifiers for each method. The rule extraction methods *LAP* and *RuleVI* are only applied to the best neural network learnt. The size of the final neural network (after training and pruning) that was chosen for rule extraction (based on the best classification accuracy and the lowest Mean square error) was 31: 1: 1 (input: hidden: output nodes).

Table 1 shows that some of the instances are incorrectly classified. The classification accuracy is still quite high considering that data comes from an on-line collection. Analysis of the results shows that most of the misclassified patterns belong to the *Risky* class. One of the reasons of low accuracy in all classifiers is uneven distribution of objects that represent *Risky* and *Safe* cases in the QR data, and some noise present in the data. Also the QR data is an example of a non-separable problem (non-disjoint distribution of target classes). This poses a problem for machine learning tools to distinguish between the two classes.

Table 1: Performances of Different Classifiers

Classifiers	Classification Accuracy		Number of Rules
	Training	Testing	
ANN	93.46	91.9	-
ANN-LAP	93.46	91.9	17 (8,3)
ANN-RuleVI	93.01	91.9	18 (10,8)
C5	94.1	92.5	15 (7,8)
FOIL	92.31	76.3	9 (9,0)

Table 1 also shows the number of generated rules. Numbers in the brackets next to the total number of rules indicate the number of rules belonging to *Safe* and *Risky* classes respectively. Some of the attributes that appear in rule-sets to state an accident-prone level crossing are: *Protection = nil, Road-visibility = poor, Train-speed = fast or very-fast (50-160 km/h), Pedestrian = exist, Approach-sign = yes, etc.*

An example rule generated by C5 is:
If Rail Visibility = poor & Train-speed = very-fast & Intersection = right-angled Then an accident may occur.

An example rule generated by GYAN is:
If Protection (Gate) = none & Pedestrian density = high & Approach-sign = yes Then an accident may occur.

The results show that C5 yields better accuracy (differing with a very small amount only) than the GYAN methodology (ANN based). The results confirm that symbolic methods (C5) are more suitable for *sequential tasks* (where the relevance of a particular input variable depends on the values of other input variables) than connectionist methods (GYAN). The results also show the poor accuracy (both for the seen and unseen instances) for the FOIL system. The results confirm that if a problem is efficiently learned by a propositional learner, a first order inductive learner may not be a good choice. The results also reveal that the first-order learning systems (such as FOIL) show a serious degradation of performance when moving from training examples to test data.

Finally, rules obtained from various classifiers are able to reveal why a particular object is classified as an accident-prone case. The classifiers also analysed which attributes (and the values) are responsible to cause accidents.

5. TECHNICAL PROBLEMS IN INTELLIGENT DATA ANALYSIS AND THEIR RAMIFICATIONS

There are many obstacles in applying IDA methods to real-world problems including lack of efficient and automatic pre-processing tools, lack of tools suitable for large, rich and complex data sets, lack of user friendly and effective post processing tools, and lack of a truly integrated data analysis environment. Following is the discussion of some of the problems may appear during a data analysis process and their suggested solutions.

Data Volume
With advances in data collection methods, data to be analysed is typically large in volume. The data set can be large in terms of number of patterns/cases/records/tuples or number of variables/features/attributes/fields. IDA methods must be scalable accordingly, e.g., (1) If a method works well for a task involving thousands of patterns, then it should work well for one with millions of patterns, and (2) If a method is successfully applied to a task involving dozens of variables, then it should be effectively applied to a task with hundreds of variables. Data analysis methods must perform satisfactorily on such large volume of data.

Enumeration of all patterns and variables may be expensive and not necessary. In spite, selection of representative patterns that capture the essence of the entire data set and their use for analysing the data set may prove a more effective approach. But then selection of such data subset becomes a problem. A more efficient approach would be to use an iterative and interactive technique that takes account into real time responses and feedback into calculation. An interactive process involves human analyst in the process, so an instant feedback can be included in the process. An iterative process first considers a selected number of attributes chosen by the user for analysis or using a feature selection algorithm, and then keeps adding other attributes for analysis until the user is satisfied. The novelty of this iterative method will be that it reduces the search space significantly (due to the less number of attributes involved). Most of the existing techniques suffer from the (very large) dimensionality of the search space [11].

There is a significant advance in agent technology. Today, agents exist to find and summarise the relevant information on the web or news feeds or other real data streams with user defined search profile [4]. Data analysis techniques can use agent technology to solve large data sets problem by contracting with a number of agents. Each agent will act independently such as identifying, accessing and storing relevant data, bidding for the work and delivering a piece of the overall solution in conjunction with other agents.

Data Quality
One major source of difficulties for data analysis methods is data quality. The data may contain noise, incomplete information and redundant and useless data. Noisy, corrupt and incomplete data can misguide the search, and makes analysis harder. However quality of data is increased with the use of electronic interchange as there is less space for noise due to electronic storage rather than manual processing.

Data analysis methods must provide adequate mechanism for finding accurate results from noisy data. Data analysis methods must facilitate both the selection of relevant data, and learning with incomplete knowledge.

Data pre-processing methods should be applied in a given situation. The procedure to ensure quality in the data must be an efficient one, otherwise may result in inappropriate data processing [6]. Methods of evaluating the usefulness of the pre-processed data are important. A domain expert should also be included in the process if possible. Usually, the data pre-processing step is application oriented, and hard to take benefit from previous research. Some of the researches indicate that it is possible to develop data pre-processing tools to be customised and used in different applications [10].

Another solution is to integrate the database technology such as data warehousing that provides a capability for the (good quality) data storage. A warehouse integrates data from multiple and heterogeneous operational sources and handles issues such as data inconsistency, missing values, etc before storing a detailed data.

Data Format
For the last decade or so, the format of data to be analysed is varied dramatically. There are many kind of data available for analysis such as relational, object-oriented, text, temporal, spatial, combinatorial, web, XML, multimedia.

This type of data requires additional steps before applying to traditional IDA models and algorithms, whose source is mostly confined to structured or text or numbers data. This additional step includes transforming advanced data format to a format suitable for traditional IDA methods.

For example, data collected from advanced applications such as web-enabled e-business sources is semi-structured and hierarchical, i.e. the data has no absolute schema fixed in advance, and the extracted structure may be irregular or incomplete [1]. Query languages can be used to obtain structural information from semi-structured data. Based on this structural information, data appropriate to traditional IDA methods are generated. Web query languages that combine path expressions with an SQL-style syntax such as Lorel or UNQL are a good choice for extracting structural information [1].

Data format can be of XML since it is assumed that in few years XML will be the most highly used language of Internet in representing documents. Assuming the metadata stored in XML, the integration of the two disparate data sources becomes much more transparent, field names can be matched more easily and semantic conflicts may be described explicitly [1]. As a result, the types of data input to and output from the learned models and the detailed form of the models can be determined. Moreover, many query languages such as XML-QL, XSL and XML-GL are designed specifically for querying XML and getting structured information from these documents. Still there are major issues to resolve such as how to use the extracted generalised DTD structure information in data analysis, how to use metadata stored in XML in data analysis, how to fill missing information if there is mismatch in attributes and others.

Sometimes much of the data of an organisation is not in simple numbers and text but in other media such as images or audio. The technology to support indexing and searching of images, sound files, and video must be used to pre process this type of data. These technologies are in progress but immature.

Data Adaptability
Data analysis systems should be adapted to deal with real-time data in which new transaction data is incorporated for analysis, and also to incorporate new data analysis models and algorithms.

Data analysis systems should take advantage of newly acquired information that was not previously available when knowledge was extracted, and combine it with the existing data model. For example, each time a new product is introduced, the company must learn a new set of best practices.

With new applications such as multimedia, XML, etc, new algorithms are developed (or the existing algorithms are updated) to deal with such data. Existing data analysis systems that include techniques to analyse the simple numbered data should also be flexible enough to include techniques to analyse the advanced type of data.

The solution can be to dynamically modifying analysed information as the dataset changes or to incorporate user feedback to modify the actions performed by the system. User-interface agents can be used to try to maximize the productivity of current users' interactions with the system by adapting behaviours.

Knowledge Representation
Information gained from the derived data model should be understandable/interpretable to users, and ultimately to be useful in decision making. The interfacing between the output of the data modelling process and representation tools will have to be transparent. For example, results of neural networks based analysis need to be represented to users in an understandable format such as symbolic rules, not just in mathematical equations.

There must be some efforts to provide standard application programming interfaces (APIs) to support extracted knowledge base interoperation. Data analysis community should also be involved with XML and related protocols and standards for standard representation, since it is predicted that in few years XML will be the data exchange language.

Knowledge Evaluation
When an IDA method is applied to a data set, it usually yields a number of result sets, especially if cross-validation and voting techniques are used in analysis. The question arises – which one to report or utilise in decision making.

The derived knowledge should not just be judged based upon accuracy. There should be some measures to estimate how effective, useful, interesting and understandable or interpretable the analysed information is. Since measures for performance evaluation depend on the learning task, the application domain should play a major role in deciding evaluation criteria. For example, in medical diagnosis it is crucial that any computerized system is able to explain and justified its decisions when diagnosing a new patient along with the high accuracy of results. When evaluating the quality of a data model in a medical domain, some of the measures are classification accuracy sensitivity, specificity, post-test probability and misclassification cost [9].

Selection of an IDA method There are always a number of methods available that can be successfully applied to underlying application problem. The question arises – which one should be used?

Appropriateness of a suitable technique to a data set should be based on certain factors such as characteristics of the data set, strength and weakness of each individual method under consideration, availability of resources, usefulness of the concluded result and some previous but recent comparative studies on effectiveness of various methods.

Recently according to several researchers, an ensemble of data analysis methods (classifiers) can often yield more accurate results than any individual method (classifier) alone. The integration of classifiers can be made statically, as dividing a problem into sub-problems, or dynamically, taking into account the variables of the new pattern during analysis.

6. CONCLUSION

Data analysis is typically an iterative and interactive process involving problem formulation, ensuring data quality, model construction, and interpretation and post-processing of the results. This paper explored the issues involved and outstanding problems in intelligent data analysis.

This paper also presented an integrated toolkit to perform the data analysis tasks and successfully demonstrate with a real-life example of the risk assessment of level crossing data. We first showed how a neural network based technique can successfully be applied to perform data analysis tasks. We then extended the analysis tool to include rule induction techniques. By including different types of machine learning techniques, this toolkit gets advantage of applying it to various data analysis tasks according to their type. The empirical results on the level-crossing data demonstrate that machine learning techniques can be applied to discover hidden information from data with high accuracy and good comprehensibility.

Many software vendors and publications are forecasting that all knowledge workers will become data analyst in the future. But still, sophisticated tools such as neural networks, decision trees and data visualisation widely available to naïve users may be a mistake. Data analysis systems should be developed as integrated systems with the needs of non-technologists end-users in mind - hiding the details of the underlying techniques, and providing an effective and user friendly interfaces. The focus should be at the process as a whole rather than at individual components in the data analysis process. We need the data analysis systems that incorporate pre-processing tasks (data cleaning, transformation, etc), multiple discoveries tasks (classification, clustering, etc), and post processing tasks (visualization) in their environment.

For success, data analysis techniques have to be coupled with (1) data management technology to capture the data in an organised fashion to systematically begin the process, (2) effective, yet simple user interface technology to enable the analysed knowledge representation. For example, a data file containing query results from a data warehouse is input to data analysis tools. The outputs of data analysis go though some post processing tools such as visualisation or rule-processor for meaningful interpretations. These modified results can be made available through an Intranet/Internet to a broad group of users by client-server technologies.

7. REFERENCES

[1] S. Abiteboul, P. Buneman & D. Suciu, Data on the Web: From Relations to Substructured Data and XML, California: Morgan Kaumann, 2000.

[2] R. Agrawal & R. Srikant, Fast Algorithms for Mining Association Rules, IBM Research Report RJ9839, IBM Almaden Research Center, 1994.

[3] P. Cabena, P. Hadjinian, R. Stadler, J. Verhees & A. Zanasi, Discovering Data Mining from Concept to Implementation, Prentice Hall PTR, 1997.

[4] F. Dignum & U. Cortes (Eds.), Agent-Mediated Electronic Commerce III: Current Issues in Agent-Based Electronic Commerce Systems, Lecture Notes in Artificial Intelligence, Springer Verlag., 2001.

[5] S. E. Fahlman and C. Lebiere. The cascade-correlation learning architecture. In Advances in neural information processing systems, volume 2. Morgan Kaufmann, 1990.

[6] A. Famili, W. Shen, R. Weber & E. Simoudis, Data Preprocessing and Intelligent Data Analysis, In Intelligent Data Analysis, 1(1), 3-23, 1997.

[7] J. Han & M. Kamber, Mastering Data Mining, San Francisco: Morgan Kaufmann, 2001.

[8] D. J. Hand, Intelligent Data Analysis: Issues and Opportunities, In Intelligent Data Analysis, 2(3), 67-79, 1998.

[9] N. Lavrac, Selected techniques for data mining in medicine, Artificial Intelligence in Medicine (16) 1 pp. 3 – 23, 1999.

[10] X. Liu, Intelligent Data Analysis: Issues and Challenges, Knowledge Engineering Review, 11(4), 1996.

[11] T. M. Mitchell, Machine Learning, the McGraw-Hill Companies, Inc, 1997.

[12] R. Nayak, Gyan: A Methodology for Rule Extraction from Artificial Neural Networks. PhD thesis, QUT, Brisbane, Australia, 2000.

[13] R. Nayak, Data Mining for Web-Enabled Electronic Business Applications, to be published in Architectural Issues of Web-Enabled Electronic Business, Shi Nansi Ed., Idea Publishing Group, April 2002.

[14] R. Nayak, J. Diederich and F. Maire, "Inductive Knowledge Acquisition using Feedforward Neural Networks with Rule-Extraction", Proceedings of the Pacific Rim Knowledge Acquisition Workshop, Singapore, Nov 21-27, 1998, PP 74-86.

[15] J. R. Quinlan, Comparing connectionist and symbolic learning methods. In S. Hanson, G. Drastal, and R. Rivest, editors, Computational Learning Theory and Natural Learning Systems: Constraints and Prospects, pages 445–456. Cambridge, MA: MIT Press, 1993.

[16] J. R. Quinlan, C5, http: //www.rulequest.com.

[17] J. R. Quinlan and R. M. Cameron-Jones, Induction of logic programs: Foil and related systems, *New Generation Computing*, 13: 287–312, 1995.