International Journal of Production Research,
Vol. 44, Nos. 18–19, 15 September–1 October 2006, 4175–4191

Taylor & Francis
Taylor & Francis Group

Data mining: manufacturing and service applications

A. KUSIAK*

Intelligent Systems Laboratory, Mechanical and Industrial Engineering,
2139 Seamans Center, The University of Iowa, Iowa City, Iowa 52242–1527, USA

In this paper basic concepts of machine learning and data mining are introduced. Machine learning algorithms extract knowledge from diverse data bases that can be used to build decision-making systems. For example, based on the operational engineering data, equipment faults can be detected, the number of items to be ordered can be predicted, optimal control parameters can be determined. A framework for organizing and applying knowledge for decision-making in manufacturing and service applications is presented. The framework uses decision-making constructs such decision tables, decision maps, and atlases. It offers a new data-driven paradigm of importance to modern manufacturing and service organisations. Examples of data mining applications in industrial, medical, and pharmaceutical domains are presented. It is envisioned that the data-driven framework presented in the paper will enhance these applications.

Keywords: Data mining; Decision making; Knowledge structuring; Process modelling; Industrial applications

1. Introduction

The growing volume of enterprise data raises many challenges, with one being able to extract, store, organize, and use the knowledge generated from data sets. The data content is frequently determined by the legacy systems deployed at various applications and time periods. Understanding the meaning and structure of the stored information is difficult, as the data is often heterogeneous and distributed. The data can be modelled as continuous functions, qualitative relations, decision-making rules, and so on. The underlying physical, biological, or organizational principles dictate the form of heterogeneous models that most often interact with each other.

Despite this complex reality, decision-making is frequently perceived as illustrated in figure 1, where a set of previously collected data is processed with a decision-making tool.

Any decision-making tool needs to be versed in current data; otherwise the decisions made are not relevant. A shortcoming of a stand-alone tool is that the data in support of decisions is limited in scope and therefore the decisions made could be locally optimal.

*Email: andrew-kusiak@uiowa.edu

International Journal of Production Research
ISSN 0020–7543 print/ISSN 1366–588X online © 2006 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/00207540600632216

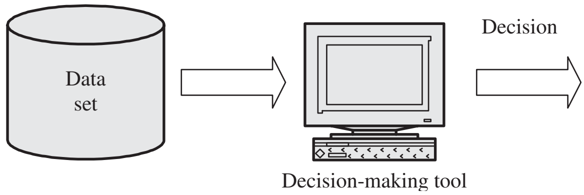

Figure 1. Stand-alone decision-making.

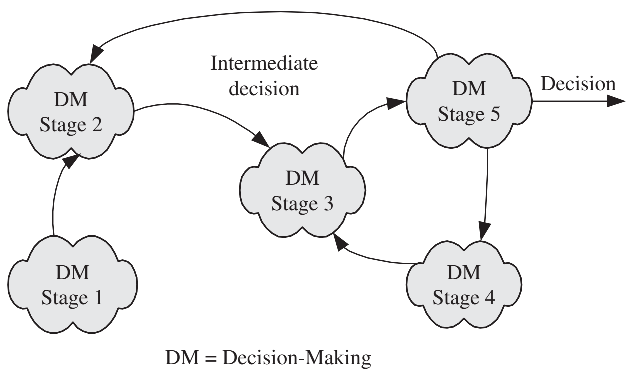

Figure 2. Distributed decision-making.

The use of tools in decision making has not been prolific due to their limited linkage with data. The developments in computing infrastructure, increased availability of data, and the emergence of data mining tools have shown a profound impact on informed, transparent, and autonomous decision-making. The decision-making environment no longer involves isolated 'islands of decisions' (somewhat analogous to the islands of automation), rather it tends to be an integrated process, distributed in space and time as illustrated in figure 2.

The progress made in developments of data analysis tools and networking technology has eliminated the boundaries faced by the decision-making systems. The data used in decision making is not only obtained from different sources but may involve different modalities (e.g. images, numbers, and personal assessment) and therefore the decision-making can be distributed. Various decision-making tools can be networked (see figure 2) in support of globally optimal decisions.

Data mining will play a dominant role in the extraction of knowledge from the distributed and heterogeneous sources of data. Different scenarios of distributed data mining are possible, for example:

- Knowledge extraction may be distributed with intermediate and final decisions generated at single or multiple sites.
- The locally generated outcomes may impact the downstream data sets or serve as intermediate decisions determining the next decision stage.
- The flow of data may form cycles rather than being linear.

Understanding and automation of a decision process in such distributed and heterogeneous data environment calls for new tools for the definition of the information content (features and decisions), data flow, and structuring knowledge used for decision-making.

In the next section data mining applications that have motivated this paper are briefly discussed.

2. Illustrative data mining applications

Illustrative examples of manufacturing and service applications where data mining have been successfully applied are presented next.

2.1 *Aluminium processing*

Aluminium and metal processing industries are interested in developing a system for the delivery of high quality products. Though advanced control systems, including digital and neural-network controllers, may be involved in the process, the values of some parameters are set by operators, e.g. the set values of the controllers. Data mining has been successfully used to develop a meta-controller seamlessly integrating all process parameters (Kusiak 2002).

2.2 *Semi-conductor manufacturing*

Semi-conductor manufacturers are interested in improving quality of wafers. The manufacturing environments at different wafer producing plants may be similar, however, for unknown reasons some wafers attain perfect quality and while other batches are not acceptable. Even within individual plants product quality varies. Data mining has been used to determine the ranges of control parameters leading to the production of wafers of the desired quality (Kusiak 2001).

2.3 *Electronic assembly*

Electronic product assembly lines face a quality problem with printed-circuit boards where assemblies rather than components fail the quality test for often unknown reasons. The management is not satisfied with the current process control or other tools that provide solutions for a ‘population’ of products rather than an ‘individual’ product. However, predicting circumstances under which an individual product (object) might fail, and thereby prevention of the failure is of great interest to the semi-conductor industry. Data mining has been applied to determine the conditions of process parameters minimizing the production of faulty products (Kusiak and Kurasek 2001).

2.4 *DNA manufacturing*

DNA manufacturing is one of the most recent processes developed for a product (DNA) that did not need to be designed, rather adopted from nature. The DNA manufacturing process, which has evolved from a biological laboratory to an industrial-scale process, involves many unknowns. Product quality and consistency is an issue that can be addressed with data mining.

2.5 *Biotechnology and chemical process industry*

Processes in the two industries are ideal candidates for data mining applications. They usually involve control spaces that cannot be effectively modelled with the classical tools, yet bio- and chemical processes could be well equipped in data collection systems.

2.6 *Energy production*

Energy is of paramount important to manufacturing and economy in general. The energy producing systems are some of the best instrumented for data collection. One of many possible fault prevention applications of data mining in the combustion process is discussed in Kusiak *et al*. (2005).

2.7 *Medical applications*

The expenditure on healthcare is growing and in some countries is beginning to exceed the manufacturing output. Systems are needed to collect the vast amount of data, analyse it, and use it for decision support in various healthcare domains (Kusiak *et al*. 2001, 2005, 2006, Shah and Kusiak 2004). Clinical and non-clinical data needs to be fused into integrated applications. One of the positive developments in healthcare is the growing use of concepts, techniques, and tools developed in manufacturing, e.g. lean manufacturing, process modelling, layout design, group technology, process planning, and quality engineering. Many fall short of equating a healthcare system to a manufacturing system.

2.8 *Pharmaceutical applications*

The scope of pharmaceutical applications is large and it may involve drug manufacturing processes as well as data processing. Data processing and analysis is a key area in the pharmaceutical industry. The vision of a pharmaceutical industry that can be achieved with data mining is shown in figure 3.

One will soon see pharmaceutical companies delivering drugs, developing test kits (including genetic tests) and computer programs to deliver the best drug to the patient. The current data mining tools are capable of issuing customized prescriptions, providing the most effective drugs and dosages with minimal adverse effects. It has become practical to think of designing, producing, and delivering drugs intended for an individual patient (or a small population of patients). Here, the analogy of one-of-a-kind production of today versus mass production is worthy of attention. Many industries are attempting to produce customized products (Agard and Kusiak 2004) and the pharmaceutical industry may soon become a part of this (Kusiak and Shah 2006).

The eight representative applications of data mining often lead to the same question: Can one determine settings of parameters (features) resulting in a desirable outcome based on the extracted knowledge from vast data sets?

This feature subset can be constructed from different data sets and processed by different algorithms. Such a subset of features and their values constitute a decision signature (defined in Kusiak 2002). The decision signature for a test engineer

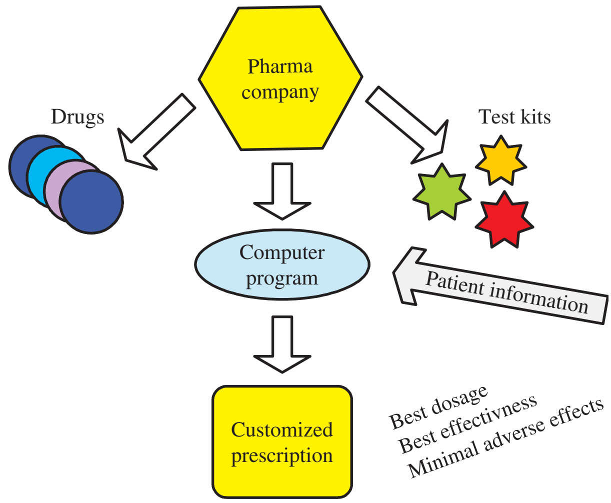

Figure 3. The vision of pharmaceutical industry.

becomes a fault signature; for a process controller a control signature, for a physician, a disease signature, and so on. Research is needed to identify decision signatures and applying them to solve production quality problems, to find the best operating conditions for a biotechnology process, and find other answers. If there were just a single decision signature for an application, there would be little room for the research reported in this paper. The fact is that decision signatures are highly individualized, and therefore the problem of generating and using these signatures for decision making needs research attention. In many applications decisions have to be reached quickly and therefore the decision signatures can be selected and evaluated before their implementation. Often, the decision signatures have to be evaluated using multiple criteria, e.g. utility and accuracy.

3. Knowledge extraction

The growing interest in data mining has led to the development of many algorithms that extract knowledge (here rules) and features from large data sets. These impressive developments in data mining have not been matched by the progress in decision making based on the knowledge extracted by machine-learning algorithms. The latter may be explained by the fact that most data mining results have been developed by the non-engineering community and focus on the knowledge extraction process rather than decision making.

Bazan (1998) categorized learning algorithms of particular interest to industry in four categories:

- Decision tree algorithms (Friedman *et al.* 1996, Quinlan 1986).
- Decision rule algorithms (Clark and Boswell 1989).
- Inductive logic programming algorithms (Michalski *et al.* 1986).
- Rough set algorithms (Grzymala-Busse 1997, Pawlak 1982).

3.1 *Metrics for evaluation of features*

Data sets and features can be directly evaluated by the following metrics:

- Upper and lower approximation measures (defined in Pawlak 1991).
- Classification quality (Pawlak 1991).
- Entropy measure (Quinlan 1986).
- Gini index (Breiman *et al*. 1984).
- Correlation, distribution type, and so on.
- Other metrics such as percentage of missing values, data error, and discretisation parameters.

Classification accuracy of the knowledge extracted from a data set is the most widely used indirect method of feature evaluation. However, this method is computationally expensive as demonstrated in De Jong (2006).

This paper fills the gap in decision making that has been created by the unprecedented progress in data mining. A universal decision-making framework and algorithms are presented for an effective use of the knowledge extracted by data mining algorithms. Numerous application areas, e.g. equipment diagnostics, quality engineering, medical diagnosis, planning, and forecasting, await decision-making tools as the volume of data collected increases at a rapid pace.

The knowledge extraction process, in its most common form, leads to decision rules (IF ... THEN rules) derived from a data set as illustrated in example 1.

Example 1
Consider the data set for eight objects (examples), each representing symptoms of a product quality, denoted here as features F1–F4 (see figure 4). Each feature (parameter) has a value, e.g. feature F3 takes values, Yes = discoloration present, 0 = discoloration not present. The decision outcome D = {Accept, Reject}.

The data in figure 4 is called a training data set. This set is transformed in decision rules by a rule extraction algorithm. Examples of rule sets in two different forms are presented in figures 5 and 6. The clause values for the rules in figure 5 are in the form of inequalities and equality, while the values in figure 6 are presented as intervals and equality.

The decision rules represent all objects (examples) of the training set. For example, decision rule R1 in figure 5 is associated with three objects [1, 3, 7] in figure 4, i.e. the value of feature F3 = Yes uniquely identifies each of the

ID	F1	F2	F3	F4	D
1	1.02	0.05	Yes	2.03	Accept
2	1.03	3.04	No	1.01	Reject
3	2.01	0.95	Yes	1.97	Accept
4	2.03	2.05	No	3.01	Accept
5	0.03	1.97	No	2.02	Reject
6	0.04	1.05	No	1.04	Reject
7	0.99	3.04	Yes	1.04	Accept
8	1.02	0.97	No	3.01	Reject

Figure 4. Training data set for the product quality problem.

Decision rule R1. `(IF F3 = Yes) THEN (D = Accept); [1, 3, 7]`
Decision rule R2. `(IF F1 >= 1.53) THEN (D = Accept); [3, 4]`
Decision rule R3. `(F1 <= 1.52) AND IF F3 = No) THEN (D = Reject); [2, 5, 6, 8]`

Figure 5. Rules involving inequality and equality relationships extracted from the training data in figure 4.

Decision rule R1. `(IF F3 = Yes) THEN (D = Accept); [1, 3, 7]`
Decision rule R2. `(IF F2 in [2.01, 2.55]) THEN (D = Accept); [4]`
Decision rule R3. `(IF F1 in [1.02, 1.52]) THEN (D = Reject); [2]`
Decision rule R4. `(IF F1 in [0.03, 0.52]) THEN (D = Reject); [5, 6]`
Decision rule R5. `(IF F2 in [0.96, 1.01]) THEN (D = Reject); [8]`

Figure 6. Rules involving inclusion and equality relationships extracted from the data set in figure 4.

three objects. Note that the number of features used to recognize all eight objects of figure 4 is two, F1 and F3. This demonstrates the feature reduction aspect of data mining. In the rule set in figure 6 three features F1, F2, and F3 are used.

In the two different rule sets, some objects may be identified by more than one rule. For example, rule R1 and R2 from figure 5 and rule R1 from figure 6 represent object 3. The type of rule extraction algorithm used and the number of features included in the rule set impact the number of rules describing a particular object in the training set.

The decision rules derived from training data sets are used to make decisions for objects (cases) with unknown outcomes. For example, assume that for the features in (1) the following data is provided

$$\begin{array}{ccc} \texttt{F2} & \texttt{F3} & \texttt{D} \\ \texttt{2.4} & \texttt{Yes} & \texttt{?} \end{array} \tag{1}$$

As the values 2.4 and Yes of features F2 and F3 match the condition values of the rules R1 and R2 in figure 6 the decision `D = Accept` is assigned to the object in (1).

Rather than making decisions directly with the decision rules, an approach that is more structured and transparent to the user is discussed in the next section.

4. Decision-making

The complexity of decision-making with rules derived by the machine learning algorithms, increases with the width of data sets. To cope with such a high degree of complexity, optimising decision tradeoffs, and making the decision process transparent to the user, new concepts of a decision table, decision map, and a decision atlas are proposed. These concepts are useful in knowledge management and structuring of corporate memory as discussed in Huang *et al.* (2005). To introduce these new concepts consider the matrix in figure 7 representing the rule set from figure 6.

	F1	F2	F3	D	Supporting examples
R1			Yes	Accept	1, 3, 7
R2		[2.01–2.55]		Accept	4
R3	[1.02–152]			Reject	2
R4	[0.03–0.52]			Reject	5, 6
R5		[0.96–1.01]		Reject	8

Figure 7. Matrix representing the decision rules from figure 6.

	F1	F2	F3	D	Supporting examples
R4	[0.03–0.52]			Reject	5, 6
R3	[1.02–1.52]			Reject	2
R5		[0.96–1.01]		Reject	8
R2		[2.01–2.55]		Accept	4
R1			Yes	Accept	1, 3, 7

Figure 8. Structured matrix derived from the data in figure 7.

To facilitate effective decision-making, the matrix in figure 7 is transformed into the matrix in figure 8.

Matching the object in (1) with the rules, now in figure 8, is more user friendly than dealing with the decision rules in figure 6.

The structured matrix in figure 8 becomes a basis of the decision table proposed in this paper. Multiple decision tables are combined in maps and those in turn in an atlas. One of the clear advantages of the structured matrix in figure 8 is the ease of generating the decision for the object in (1). Sliding this object through the matrix in figure 8 matches this object with rulesR1 and R2, thus resulting in the decision `D = Accept`. The support for this decision with the examples 1, 3, 4, and 7 of the training set from figure 4 is clearly visible. Any additional information needed for decision-making (e.g. risks, pointers to alternative rule sets) may be conveniently associated with each rule in figure 8. Cluster analysis algorithms are suitable for structuring rule-feature matrices.

4.1 *Decision tables, maps, and atlases*

A decision table is a collection of knowledge needed to make decisions in a particular area (see figure 9). It generalizes the concept of the structured matrix introduced in figure 8 by

- including decision rules generated by different machine learning algorithms;
- post-processing rules, e.g. combine rules for improved interpretability;
- content structuring, e.g. accomplished with a clustering algorithm.

The entries of the decision table in figure 9 are the attribute values generated by machine learning algorithms. They may represent singular numerical or symbolic values, bounded ranges of values, unbounded value ranges (inequalities), and so on.

Decision basis and justification are also defined in the table (see figure 9). Each decision basis contains feature values, while the justification characterizes the decision basis, e.g. it may contain the strength of the decision basis and risks

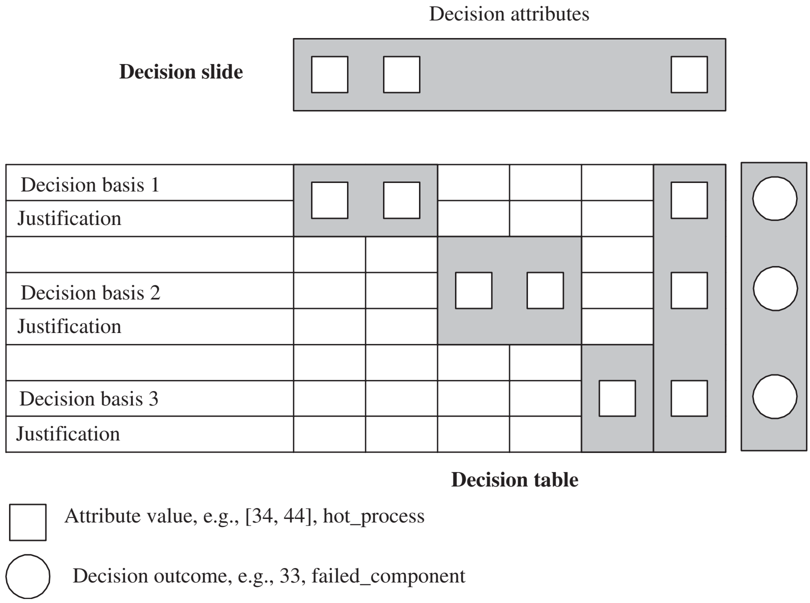

Figure 9. The concept of a decision table and a slide.

associated with a decision. The actual decision can be made using the content of one or more decision tables at any step of the decision map (see figure 10). One way is to match the content of the decision slide containing selected attribute values of an object with unknown outcome to an appropriate row of the decision table, e.g. decision basis 1 may correspond to rule R1 in figure 5. The new object is assigned an outcome equal to the outcome of the matching decision basis.

Decision-making is often distributed over numerous steps (see figure 2 and figure 10), each involving one or more decision tables. There are two primary reasons for alternative decision tables. One being that a decision may be accomplished in various ways based on the decision tables. The second is the increased user's transparency of the results generated from variety of tables. The notion of independence has a profound impact on user's confidence in the algorithmically generated results. The decision process model and the collection of decision tables are called a decision map (figure 10). Maps in turn can be combined in an atlas and multiple atlases make up a library. The proposed constructs are suitable for the development of active and interactive web-based applications.

4.2 *Process modelling and decision-making*

The notion of a process and a modelling methodology are of importance in data-driven decision-making. The methodology selected for modelling a process depends on a number of factors ranging from organisation's culture and training to the suitability and availability of a modelling methodology or a tool. This paper does not advocate any specific process modelling methodology or a tool, rather the Integrated DEFinition (IDEF3) methodology is selected for illustrative

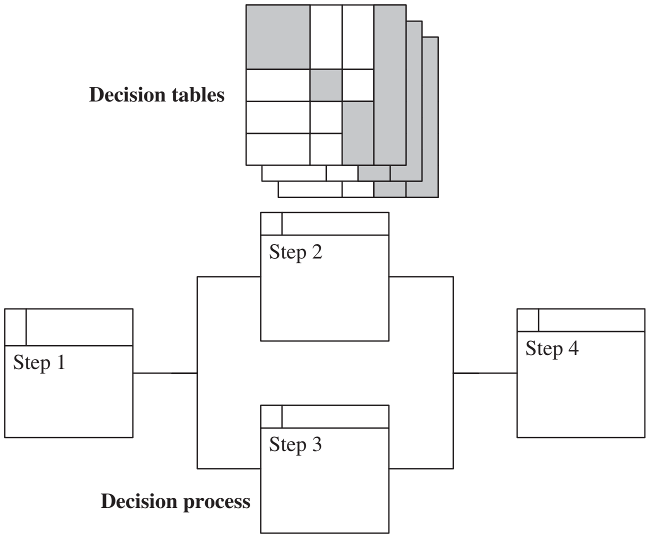

Figure 10. Decision map.

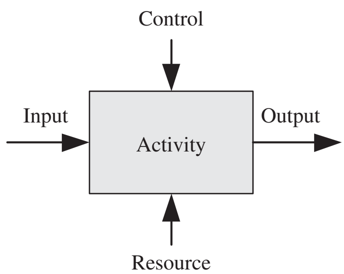

Figure 11. Basic modeling block of the IDEF3 methodology.

purposes only (Mayer *et al*. 2002). The proposed concepts can be implemented with any process modelling approach. The IDEF3 methodology is simple and widely recognized by the technology and service community. The basic block of the IDEF3 methodology (with the resource incorporated) is shown in figure 11.

In this paper, the meaning of the four arrows in figure 11 will be extended beyond the original definition provided by IDEF0 and IDEF3 as illustrated next.

Input: Data that is transformed into output.

- Independent variable, e.g. pressure.
- Input of an activity, e.g. temperature.
- Decision, e.g. quality index.

Output: The result of input transformation.

- Dependent variable, e.g. pressure.
- Output of an activity, e.g. temperature.
- Decision, e.g. predicted efficiency.

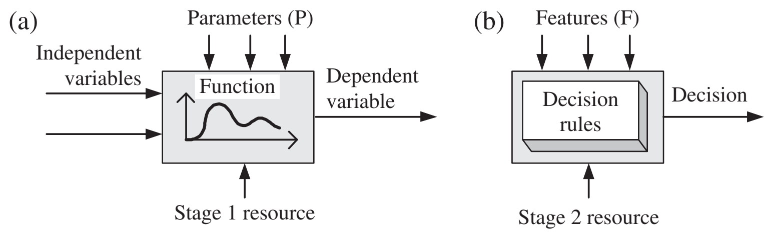

Figure 12. Examples of decision-making process blocks: (a) function block; (b) decision rule block.

Control: Necessary for the computation of the output it does not, however, undergo transformation.

- Feature (attribute), e.g. silica concentration.
- Parameter, e.g. gravity constant.

Mechanism: Resource(s) needed at a particular stage.

- Resources of many origins, e.g. chemical process, DNA manufacturing process, human decision maker, manufacturing equipment.

In this paper, the term 'Activity' of IDEF3 is replaced with 'Stage', which will have multiple meanings.

Stage: A step in the decision-making process.

Examples of different meanings of the term 'Stage' (see figure 2) are as follows:

- Activity.
- Continuous function.
- Fuzzy membership function.
- Qualitative dependency.
- Decision rules.
- Algorithm.

These expanded definitions of the basic elements of the IDEF3 methodology are illustrated with the two decision-making process blocks in figure 12.

The blocks (e.g. in figure 12) and the logical connectors (e.g. AND, OR, XOR) of the IDEF3 methodology have been used to model the decision-making process in figure 13.

4.3 *Decision-making algorithms*

An approach generating high confidence decisions is proposed. The goal is to generate $(100-\varepsilon)$ percent accurate decisions for $(100-\beta)$ percent cases with unknown decisions. The concept of orthogonal algorithms, the P (primary) and the C (confirmation) algorithm, introduced in this paper allows ε to converge possibly to 0 and β to remain small. The P-algorithm generates decisions using the structured knowledge derived by multiple learning algorithms. The C-algorithm derives the same decision using an orthogonal concept, e.g. a classical algorithm, a distance metric based algorithm. The primary decision will be derived only when the decisions of the P- and C-algorithms agree, otherwise, a secondary decision will be

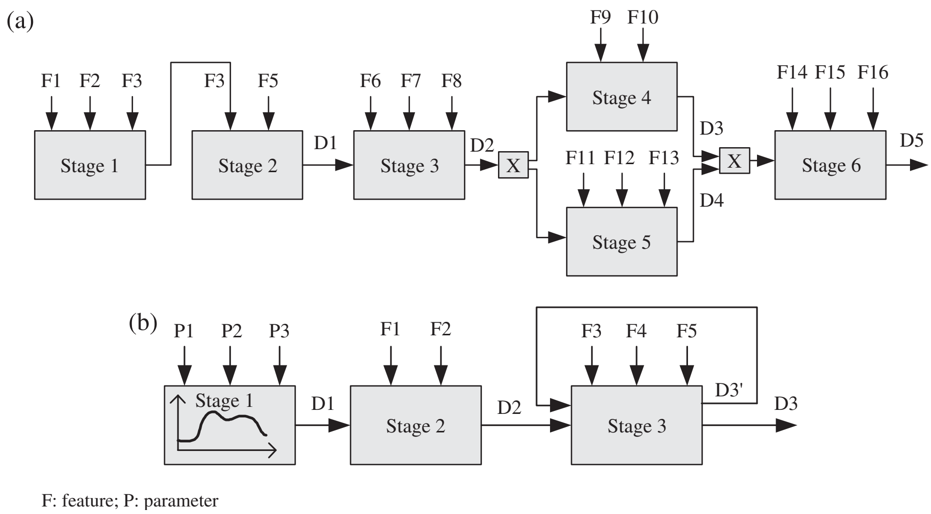

Figure 13. Decision-making process models: (a) linear process, (b) process with a cycle.

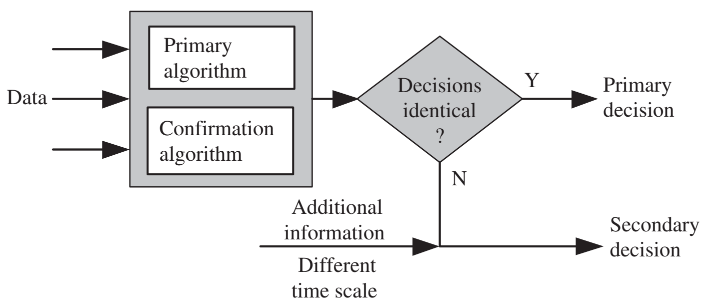

Figure 14. Decision-making logic.

generated. Note that multiple P-algorithm ($P = \{P_1, P_2, \ldots, P_p\}$) and C-algorithms ($C = \{C_1, C_2, \ldots, C_p\}$) are used. The orthogonal algorithm generalizes the bagging concept (Tan *et al*. 2006) of data mining. While the P-algorithm is to some degree parallels bagging, the C-algorithm takes decision-making to a different realm. The concept of orthogonal algorithms integrate two decision-making perspectives, one based on data mining (the P-algorithm) and the other utilising a multitude of algorithms developed in operations research and optimisation.

The interaction between the P- and C-algorithms is illustrated in figure 14.

The P-algorithm

The P (Primary) algorithm generates decisions by matching feature values of an unknown object with the extracted rules. For a new object to be matched with one rule in the basic set any of the following three cases is possible:

- The new object matches exactly one rule.
- The new object matches more than one rule.
- The new object does not match any rule.

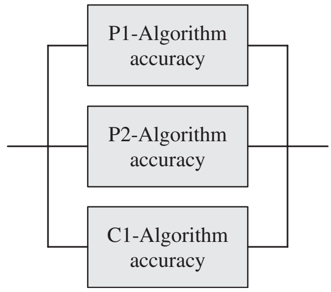

Figure 15. Systems engineering interpretation of the P- and C-algorithms.

In the presence of alternative rule sets additional cases emerge. This is a fundamental issue in decision-making with alternative rule sets.

The C-algorithm

The C-algorithm generates decisions using concepts that are 'orthogonal' to the ones applied in the P-algorithm, e.g. an algorithm based on a distance metric. The distance metric algorithm by itself might be a good tool for making decisions. This view is supported by the literature, e.g. Slowinski (1993) applied the distance metrics to make decisions based on local information.

The interaction between the P- and C-algorithms is illustrated with the systems engineering perspective in figure 15.

Each of the P- and C-algorithms can be viewed as components of a parallel system, thus contributing to the overall system accuracy. This interpretation of the two classes of algorithms makes it analogous to the system reliability perspective.

5. Data transformation

The quality of decision-making algorithms depends on the quality of the knowledge extracted from data sets. The data sets can be mined in their raw form (as collected), or they can be transformed. Constructive induction is a process of describing objects for improved classification (Wnek and Michalski 1994, Bloedorn and Michalski 1998). New features are built from the existing ones, and some features (parameters) of objects are modified or deleted. It should be noted that the deletion of features is related to the feature selection problem (Yang and Honavar 1998) which is not addressed in this paper.

The data transformation aspect can be used in manufacturing and service applications to improve usability, transparency, and the decision-making accuracy of the extracted rules.

The following data transformation approaches are widely used in data mining (Tan *et al.* 2006):

- *Filling in missing values*. For example, the most common value method replaces the missing values with the most frequent values. The data set decomposition method partitions the data set into subsets without missing values that are in turn used for mining (Ragel and Cremilleux 1998).

- *Discretisation*. For example, the equal frequency interval method groups continuous values into k intervals, where each interval contains m/k (possibly duplicated) adjacent values, where m is the number of examples (Dugherty *et al*. 1995). The recursive minimal entropy algorithm establishes intervals by considering the class information entropy (Carlett 1991, Fayyad and Irani 1993).
- *Feature content modification*. Feature generalisation and specialisation methods are discussed in Han and Kamber (2001).
- *Wavelets*. Used for analysis of temporal data sets (see Jorgensen 2003, 2006, Daubechies and Sweldens 1998).

Other data transformation methods are discussed in Cios *et al*. (1998) and Han and Kamber (2001).

6. Data mining in systems integration

The enterprise of the future is likely to be distributed in time and space and its structure will be dynamically changing. The pressure to reduce the time to integrate various enterprise systems will be growing. At the same time, a stronger degree of integration among the interacting systems will be needed. While successful standardisation efforts may alleviate some of the integration issues, the 'integration gap' is not likely to diminish. Another complicating factor is that integration does not have a natural ownership in the research and industrial community. In terms of research, integration is sometimes associated with industrial and systems engineering; however, no meaningful research results on systems integration have been published. Integration companies deal with the day-to-day business issues. There might be two basic reasons for the sluggish progress in the integration research. First of all, the fragmented integration knowledge once developed in a singular application tends to be considered as providing a technological advantage and for that reason it is not well documented and distributed outside a company's boundaries. The second reason may be the lack of an existing theory that would naturally lend itself to the integration applications. While it is difficult to change the first obstacle with a science-based approach, the second can certainly be tackled. Data mining is a natural fit for systems integration. Data mining algorithms build relationships among features of diverse systems. These relationships are difficult to capture with any other known tool. Algorithms can be developed to integrate an overall system performance with the component systems based on the extracted knowledge. The system integration will be accomplished in two phases, learning (data mining) and decision-making. The experience with the distributed scheduling systems indicates that a seamless integration of independently developed was applications is possible. The data mining algorithm provided a viable integrating mechanism. The quality of this integration mechanism could be improved by continuous learning. The learning phase allows for scalability of the approach as the number and structure of the interfacing component systems changes.

7. Data farming

Data farming is a process of defining features that are most appropriate for data mining and system integration (Kusiak 2006). In many technology and medical applications data sets are not available and the question arises as to what makes the most appropriate data set. The data collection may be limited to pure data recording or may involve installation of sensors, or other more complex measures, e.g. defining bundles of features. Prior computational results with process industry data indicate that bundles of features can significantly improve the quality of knowledge extracted from data sets.

Data farming can be contrasted to feature selection (see, for example, Bradley *et al.* 1998, Kittler 1975) that is widely used in data mining for choosing features among the ones available in the training data set. The features in the feature selection approach are known while data farming defines the appropriate set of features in terms of cost, time, accuracy, and so on.

Thus far, most data mining efforts have been based on data sets collected for unrelated purposes, e.g. routine data collection requirements, process improvement projects, or government regulations. Data mining brings two meaningful benefits to organisations, data reduction and increased economic benefits from the data. The two easily translate into cost savings and increased competitiveness. The appropriateness of features can be expressed with different measures, e.g. maximisation of the prediction accuracy, minimisation of the cost of data collection.

Measuring utility of the data collected is importance in decision making. For example, a crude measure justifying the data collection effort for a set of features can be expressed with the increase in the classification accuracy or the relative rule support. These two metrics are commonly used to characterize data sets in data mining.

8. Conclusion

The growing volume of data in manufacturing and service industries is a challenge that needs research on tools that discover unique properties of the data. Data mining has emerged as a discipline that offers tools for data analysis and knowledge discovery. Considerable progress has been made in the development of algorithms for extracting knowledge from databases. This progress has not been matched by research on decision-making algorithms, which make use of the knowledge obtained by the machine learning algorithms. This paper outlined a framework for decision making based on the knowledge provided by different data mining algorithms. The proposed decision-making approach is suitable for applications in areas such as equipment and medical diagnosis, quality engineering, scheduling, and general category of hierarchical decision making, where outcomes of different accuracy requiring diverse background information are produced.

A novel concept of decision atlas, maps, and tables was introduced. The decision atlas will increase transparency and effectiveness of the decision-making process. Implementation of theoretical results is often as difficult as the research leading to the discovery of the result itself. This implementation barrier can be overcome by a user-friendly, transparent, and convincing presentation of justifications for the decisions. The decision tables, maps, and atlas introduced in this paper aim at meeting these important need.

Though the ideas discussed in this paper apply to many domains, they were illustrated with a class of decision-making problems in selected industries.

The fact that the proposed data mining approach involves two phases: learning and decision-making, makes data mining suitable for system-on-a-chip applications. The learning phase has high computational complexity and can be performed off-line; however, the real-time processing is also possible. The extracted decision logic may be encoded on a chip together with decision-making algorithms that have lower computational complexity.

References

Agard, B. and Kusiak, A., A data-mining based methodology for the design of product families. *Int. J. Prod. Res.*, 2004, **42**(15), 2955–2969.

Bazan, J.G., A comparison of dynamic and non-dynamic rough set methods for extracting laws from decision tables, In *Rough Sets in Knowledge Discovery 1: Methodology and Applications*, edited by L. Polkowski and A. Skowron, pp. 322–365, 1998 (Springer-Verlag: New York).

Bloedorn, E. and Michalski, R.S., Data-driven constructive induction. *IEEE Intel. Syst.*, 1998, **13**(2), 30–37.

Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, P.J., *Classification and Regression Trees*, 1984 (Wadworth International Group: Belmont, CA).

Bradley, P.S., Mangasarian, O.L. and Street, W.N., Feature extraction via mathematical programming. *INFORMS J. Computing*, 1998, **10**, 209–217.

Carlett, J., Megainduction: Machine learning on very large databases. Ph.D. thesis, Department of Computer Science, University of Sydney, Australia, 1991.

Cios, K., Pedrycz, W. and Swiniarski, R., *Data Mining: Methods for Knowledge Discovery*, 1998 (Kluwer: Boston, MA).

Clark, P. and Boswell, R., The CN2 induction algorithm. *Mach. Learn.*, 1989, **3**(4), 261–283.

Daubechies, I. and Sweldens, W., Factoring wavelet transforms into lifting steps. *J. Fourier Anal. Appl.*, 1998, **4**, 247–2002.

De Jong, K.A., *Evolutionary Computation: A Unified Approach*, 2006 (MIT Press: Cambridge, MA).

Dugherty, D., Kohavi, R. and Sahami, M., Supervised and unsupervised discretisation of continuous features, in *Proceedings of the 12th International Machine Learning Conference*, 1995, pp. 194–202.

Fayyad, U.M. and Irani, K.B., Multi-interval discretisation of continuously-valued attributes for classification learning, in *Proceedings of the 13th International Joint Conference on Artificial Intelligence*, 1993, pp. 1022–1027.

Friedman, J., Yun, Y. and Kohavi, R., Lazy decision trees, in *Proceedings of the 13th National Conference on Artificial Intelligence*, AAAI Press and MIT Press, 1996.

Grzymala-Busse, J.W., A new version of the rule induction system LERS. *Fund. Inform.*, 1997, **31**, 27–39.

Han J. and Kamber, M., *Data Mining: Concepts and Techniques*, 2000 (Academic Press: San Diego, CA).

Huang, C.-C., Tseng, T.-L. and Kusiak, A., XML-based modelling of corporate memory. *IEEE Trans. Systems, Man, Cyber.*, Part A, 2005, **35**(5), 629–640.

Jorgensen, P.E.T., Matrix factorisations, algorithms, wavelets. *Notices Am. Math. Soc.*, 2003, **50**(8), 880–894.

Jorgensen, P.E.T., *Analysis and Probability: Wavelets, Signals, Fractals, Graduate Texts in Mathematics Series*, Vol. 234, 2006 (Springer: New York, NY).

Kittler, J., Mathematical methods for feature selection in pattern recognition. *Int. J. Man-Machine Studies*, 1975, **7**, 609–637.

Kusiak, A., Decomposition in data mining: An industrial case study. *IEEE Trans. Electron. Pack. Manuf.*, 2000, **23**(4), 345–353.

Kusiak, A., Rough set theory: A data mining tool for semi-conductor manufacturing. *IEEE Trans. Electron. Pack. Manuf.*, 2001, **24**(1), 44–50.

Kusiak, A., A data mining approach for generation of control signatures. *ASME Trans.: J. Manuf. Sci. Eng.*, 2002, **124**(4), 923–926.

Kusiak, A., Data farming: A primer. *Int. J. Op. Res.*, 2005, **2**(2), 48–57.

Kusiak, A. and Kurasek, C., Data mining analysis of printed-circuit board defects. *IEEE Trans. Robot. Autom.*, 2001, **17**(2), 191–196.

Kusiak, A. and Shah, S., Data mining and warehousing in pharma industry, In *Encyclopedia of Data Warehousing and Mining*, edited by J. Wang, pp. 239–244, 2006 (Idea Group, Inc.: Hershey, PA).

Kusiak, A., Kern, J.A., Kernstine, K.H. and Tseng, T.L., Autonomous decision-making: A data mining approach. *IEEE Trans. Inform. Techno. Biomed.*, 2000, **4**(4), 274–284.

Kusiak, A., Law, I.H. and Dick, M.D., The G-algorithm for extraction of robust decision rules: Children's postoperative intra-atrial arrhythmia case study. *IEEE Trans. Inform. Techno. Biomed.*, 2001, **5**(3), 225–235.

Kusiak, A., Burns, A., Shah, S. and Novotny, N., Detection of events causing pluggage of a coal-fired boiler: A data mining approach. *Combust. Sci. Tech.*, 2005, **177**(12), 2327–2348.

Kusiak, A., Dixon, B. and Shah, S., Predicting survival time for kidney dialysis patients: A data mining approach. *Comp. Biol. Med.*, 2005, **35**(4), 311–327.

Kusiak, A., Caldarone, C.A., Kelleher, M.D., Lamb, F.S., Persoon, T.J. and Burns, A., Hypoplastic left heart syndrome: Knowledge discovery with a data mining approach. *Comp. Biol. Med.*, 2006, **36**(1), 21–40.

Mayer, R.J., Cullinane, T.P., deWitte, P.S., Knappenberger, W.B., Perakath, B. and Wells, M.S., *Information Integration for Concurrent Engineering (IICE) IDEF-3 Process Description Capture Method Report*, 1992 (Armstrong Laboratory, AL-TR-1992-0057: Wright-Patterson AFB, OH).

Michalski, R.S., Bratko, I. and Kubat, M. (Eds), *Machine Learning and Data Mining*, 1998 (John Wiley: New York, NY).

Michalski, R.S., Mozetic, I., Hong, J. and Lavrac, N., The multi-purpose incremental learning system AQ15 and its testing application to three medical domains, in *Proceedings of the 5th National Conference on Artificial Intelligence*, AAAI Press, Palo Alto, CA, 1986, pp. 1041–1045.

Pawlak, Z., Rough sets. *Int. J. Inform. Comp. Sci.*, 1982, **11**(5), 341–356.

Pawlak, Z., *Rough Sets: Theoretical Aspects of Reasoning About Data*, 1991 (Kluwer: Boston, MA).

Quinlan, J.R., Induction of decision trees. *Mach. Learn.*, 1986, **1**(1), 81–106.

Ragel, A. and Cremilleux, B., Treatment of missing values for association rules, in *Proceedings of the 2nd Pacific Asia Conference*, PAKDD'98, Melbourne, Australia, 1998.

Shah, S.C. and Kusiak, A., Data mining and genetic programming based gene/SNP selection. *Artif. Intel. Med.*, 2004, **31**(3), 183–196.

Slowinski, R., Rough set learning of preferential attitude in multi-criteria decision making, In *Methodologies for Intelligent Systems*, edited by J. Komorowski and Z. Ras, pp. 642–651, 1993 (Springer-Verlag: Berlin).

Taguchi, G. and Clausing, D., Robust quality. *Harv. Business Rev.*, 1990, **January/February**, 65–75.

Tan, P.N., Steinbach, M. and Kumar, V., *Introduction to Data Mining*, 2006 (Addison Wesley: New York, NY).

Wnek, J. and Michalski, R.S., Hypothesis-driven constructive induction in AQ17-HCI: A method and experiments. *Mach. Learn.*, 1994, **14**(2), 139–168.

Yang, J. and Honavar, V., Feature subset selection using a genetic algorithm. *IEEE Intel. Syst.*, 1998, **13**(2), 44–49.