Journal of ICT, 4, pp: 37-56

RECOGNITION DECISION-MAKING MODEL USING TEMPORAL DATA MINING TECHNIQUE

* N. M. Norwawi, * K. R. Ku-Mahamud, and **S. Deris

**Department of Computer Science,*
Faculty of Information Technology,
Universiti Utara Malaysia,
06010 Sintok, Kedah, Malaysia.

***Department of Software Engineering,*
Faculty of Computer Science & Information Science,
Universiti Technology Malaysia,
83130 Skudai, Johor, Malaysia.

E-mail: nmn@uum.edu.my., ruhana@uum.edu.my and safaai@fsksm.utm.my

ABSTRACT

An accurate and timely decision is crucial in any emergency situation. This paper presents a recognition decision making model that adopts the temporal data mining approach in making decisions. Reservoir water level and rainfall measurement were used as the case study to test the developed computational recognition-primed decision (RPD) model in predicting the amount of water to be dispatched represented by the number of spillway gates. Experimental results indicated that new events can be predicted from historical events. Patterns were extracted and can be transformed into readable and descriptive rule based form.

Key words: Naturalistic decision making, recognition decision-making, temporal data mining, rapid decision

1.0 INTRODUCTION

Decision-making in real-life tasks is usually dynamic and automatic due to the intuitive processing. Good decision-making in an emergency environment also follows the same trait where experience can distinguish the most important cues in order to achieve a rapid and accurate decision. Experience enhances the ability of the decision-maker to make rapid decisions, to make situation assessment, to recognize a typical situation and ways of reacting to problems, to focus and distinguish important cues, to select a satisfying solution and mentally simulate to ensure that it is workable, to form expectancies, to detect anomalies and to describe a plausible explanation for unusual events (O'Shaughnessy, 1999;Lipshitz *et al.,* 2000; Deitch *et al.,* 2001).

In the traditional decision-making model or the normative approach, all possible solutions will be. generated. For each solution generated, weight or risk will be calculated. The optimum option will be chosen as a solution. The main idea in this classical approach is the evaluation of alternatives and then optimization. However, a considerable time is needed for evaluating every single option and comparing them against each other. Thus a normative approach is unsuitable for urgent decision-making such as in an emergency situation where timeliness is the primary concern. A faster decision-making model is needed that can facilitate rapid response and accurate decisions in such a situation.

In an emergency domain, rapid response and accurate decisions are very crucial where safety of the victims is the prime concern. However, emergency is a temporal continuous environment where there is a time-delay between cause and effects. Although studies have been done on computational recognitionprimed decision (RPD) model, no information is available on the application of RPD model in a temporal environment and the use of data mining technique.

The objective of this study is to develop a computational RPD model that incorporates the temporal data mining technique that can be used for decision making in the emergency situation. The results of this study may provide an alternative approach to implement the computational RPD model that can provide rapid and accurate decisions. This is an experimental and applied research. Reservoir flood control is taken as a test case where real reservoir operation data from 1998 to 2002 were used to test the computational RPD model developed. In this domain, reservoir water level and rainfall measurement at six rainfall stations were used to predict the spillway gate decision. The spillway gate is a flood control mechanism where water will be released from the reservoir or dam to keep the reservoir at a safe water level. The decision is critical since it involves public safety. Timeliness and accuracy of decisions is very crucial due to the unpredictability and non-linearity of Mother Nature events.

2.0 RECOGNITION-PRIMED DECISION

Klein introduced a model named 'recognition-primed decision" (RPD) to describe the decision process in the naturalistic decision making environment (Klein and Klinger, 1991; 0' Shaughnessy, 1999; Breen, 2002). The RPD model was developed based on the results of a cognitive task analysis of firefighter commander decision-making in actual fire emergency situations Klein and Klinger, 1991). The model describes how humans working in an unpredictable environment make life and death dynamic decisions in the field settings. The basis of this model is situation assessment based on experience, pattern recognition and mental simulation. RPD is one of the most recognized models to describe decision-making for dynamic and uncertain situations and has been used in various applications such as spacecraft command and control decision support system (Morris and Mitchell, 1995) , pilot decision-making (Craig, 1998) and cardiac surgery critical care (Currey and Botti, 2003).

There are three variations of Klein's RPD model (Klein and Klinger, 1991) that show the hierarchy of the solution generation according to the simplicity or complexity of the cases. The first variation is the simplest RPD model where a simple or exact match is found. The situation is recognized and the action to be implemented is obvious, quick and accurate. Furthermore, the situation assessment based on the RPD model has four byproducts: expectancies, cues, plausible goals and typical actions.

In the second variation, an action selected is checked against the expectancies. If the expectancies are violated then the situation needs to be reassessed and more information is needed. On the other hand, if expectancies are not violated, then the action selected is evaluated via mental simulation to uncover problems. Selections will be modified if there are inadequacies in the action selected. This process is repeated until the inadequacies are removed and the modified action can be executed.

The third variation represents the most complex case when the selection has some flaws and is rejected in favour of the next most typical reaction where the next option will be evaluated. In this variation, if the solution chosen is unworkable, then the next possible solution will be generated. It is a sequential approach of solution selection referred to as *serial evaluation* of solution. The main feature of this solution generation is aimed at achieving a solution that is good enough and workable.

Various methods have been used to represent the computational RPD model such as case-based reasoning (Brann *et al.,* 1995), BDI agent belief structure (Norling *et al.* 2000), artificial neural network (Liang *et. al.,* 2001), later enhanced with fuzzy logic (Robichaud, 2001) and multiple long term memory model (Warwick *et. al.,* 2001). Case-based reasoning, artificial neural network, fuzzy logic, rule-based the and intelligent agent have similar characteristics. They are intelligent techniques with human touch or human-like approach, which is human behavior oriented. The adoption of the techniques is mainly due to the overlapping characteristics between the particular technique and the RPD model.

Another technique that has a close resemblance with the RPD processes is data mining. Data mining is defined as "the nontrivial extraction of implicit, previously unknown and potential useful information from data" (Frawley *et al.,* 1992). It is an emerging area of computational intelligence where decision making is based on prior data. Kusiak *et al.* (2000) claimed that data mining is a novel approach of autonomous decision-making. The autonomous characteristics can facilitate rapid decision-making.

The resemblance between the RPD model and data mining approach lies in the process of using previous knowledge to make decisions in a new situation. Both approaches have the advantage of shorter decision response time, being process oriented, readability of the result and adaptability to an evolving decision environment (Kusiak, 2002). Pattern recognition approach is also another similar characteristic between data mining and the RPD model. These patterns were extracted from past data to represent prior knowledge or in other words, experience. Then, using a similarity search, a typical situation that is similar with the past can be obtained. If this pattern does not exist then the new pattern can be adapted into the learning model of data mining. Both the RPD model and data mining approach *use prior knowledge* and *recognitional strategy* for making future decisions and are *adaptive* to new situations. Among the similar features are human-like or behavioral based decision-making methods, use of historical data to extract knowledge, use of pattern recognition techniques for the similarity check and prediction of future decisions based on previous patterns/ experience.

3.0 RESEARCH METHOD

As discussed in Section 2.0, temporal data mining is chosen as an alternative technique used to represent the RPD model for a naturalistic situation. A mapping between the RPD model and temporal data mining is carried out to identify a suitable technique to be used for the internal processing of the RPD model specifically *learning from experience* and *situation recognition.* The nature of the data used is time series. This makes the design of the leaming from experience algorithm complex where the temporal information must be captured.

In this study, 'event' is defined as the changing point of decision where there is a transition from gate closed to gate opened. A change point detection algorithm is needed to identify the change from normal situation. A sliding window technique was used to capture the temporal pattern describing a situation leading to an event (Keogh, 2001). It is a common method used to record temporal sequences due to its simplicity and intuitiveness (Norwawi, 2004). The collection of these windows represents the temporal patterns and forms the classifier.

Next, for the situation recognition algorithm, a feature bundle matching technique was chosen due to its suitability for time series data (Kusiak, 2001). Feature bundle represents a temporal pattern where the emphasis is on the coexistence of certain attributes as a decision parameter. For example, decision on water release from the reservoir follows after two days heavy rainfall. The measurement of the two days consecutive rainfalls are bundled together to represent a parameter on the rainfall pattern.

The combination of the leaming and situation recognition algorithm forms the proposed temporal data mining technique representing a computational form of the RPD model. The technique was evaluated against the real reservoir operation data. There were two phases: training for the learning algorithm and testing for the situation recognition algorithm. Data are divided into training and testing set by preserving the order of the data according to day of operation. This is to ensure the sequence in time is captured. Parameters used for the data are the daily water levels at the reservoir, daily rainfall measurements at six rainfall stations, daily change in water levels and number of gates opened. Decision predicted is the number of gates to be opened when a similar situation arises.

Performance of the algorithms were measured by comparing the predicted decision against the actual. Attributes such as true positive, true negative, false positive and false negative were calculated. Then the value of the specificity, sensitivity and accuracy were generated to represent the quality of the prediction. The performance measurement of the temporal data mining algorithm was adopted from Gryzmala-Busse *et al.* (1999), Povinelli and Feng (2002) and Dunham (2003) as shown in Table 1.

Table 1: Performance measurement

Measurement	Meaning
t_p	True positive. Number of event correctly predicted
f_p	False positive. Number of predicted event but in actual non-event.
t_n	True negative. Number of non-event correctly predicted.
f_n	False negative. Number of predicted non-event but in actual non-event.
Sensitivity (*Sen*)	The accuracy of correctly predicted event $= \frac{t_p}{t_p + f_n}$
Specificity (*Spec*)	The accuracy of correctly predicted non-event $= \frac{t_n}{t_n + f_p}$
Total Accuracy (*Acc*)	The ratio of total correct prediction $= \frac{t_p + t_n}{t_n + f_n + t_p + f_p}$

4.0 PROPOSED COMPUTATIONAL RPD MODEL

Temporal data mining is a component of Knowledge Discovery in Temporal Databases where its main concern is the formulation of algorithms that can extract temporal patterns and be enumerated from temporal databases. In order to represent a computational version of RPD model, two major issues need to be solved: how to learn from experience? ; how to recognize a situation? as illustrated in Fig. 1.

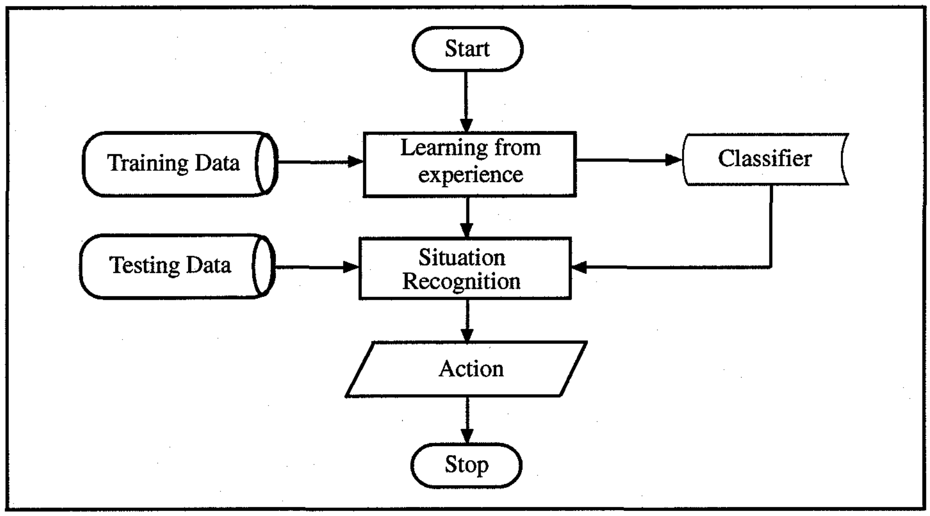

Figure 1: Major steps in temporal data mining algorithm for RPD model

The learning algorithm will store the patterns extracted from past data into the classifiers. This classifier will then be used to predict future decisions in a new situation. Fig. 2 shows the proposed RPD model based on the temporal data mining approach.

Fig. 2 shows how the temporal data mining process is embedded into the RPD model. Temporal data mining provides the experience knowledge stored in the classifier model through the learning algorithm from the feature extraction and classification steps. This classifier model will be used to check for any similarity in a new situation. A conflict resolution is also added in case a multiple match is found. An incremental learning process is also added where new cases can be added into the classifier's model.

4.1 Learning from Experience Algorithm

There are two major steps in the learning process: *feature extraction* and *classification.* Features will be extracted from past operation data. It is a batch processing activity where all possible patterns will be extracted from the training data. The patterns extracted will then be classified into separate and unique classes. Since the domain is a temporal environment, a sliding window technique was chosen for data segmentation technique for the purpose of collecting

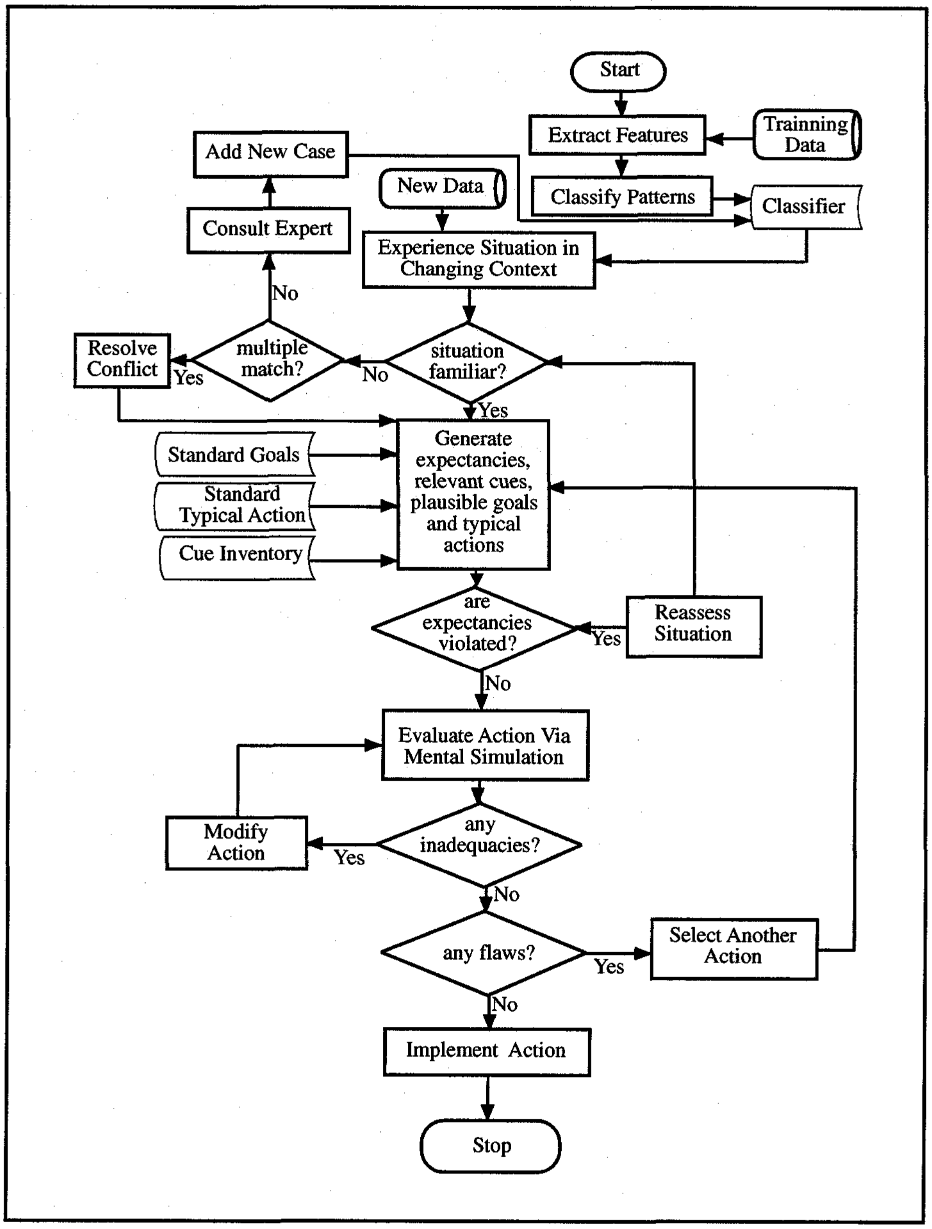

Figure 2: Proposed RPD model based on temporal data mining approach

temporal patterns to be used for the classification step (Keogh, 2001). A formal representation is discussed in Karimi and Hamilton (2002). Fig. 6 shows the learning algorithm developed.

```
function learn ( R[1..T])

 for t ← 1 to T do                         /* BATCH PROCESSING – Feature Extraction
    D[t ] ← discretize (R[t])              /* Transform data into discrete form

    i ← 1
    if true ← detect_change_point(t) then
      P_w [i] ← copy2window(D, t )         /* Form window slice  and collect window
                                            i ¨ i + 1
                                           /* Classify window

 C[1] ← P_w [1]                            /*  Set first window to the first class
 n ← 1

for j ← 2 to m do                          /* for all other window, p_2 ... p_m ,in window set P_w

   for k ← 1 to n do                       /* for all class exist
     if true ¨ class_exist(.j,k) then      /* if P_w [j] decision class does exist
    group_into class(j, k)

    C[n+1] ← P_w [j]                       /* if P_w [j] decision class does not exist add new class
    n ← n + 1

return C[1..n]
```

Figure 3: Temporal learning algorithm

The complexity of the algorithm is dependent on the number of data used. The larger the data, the more time is needed to generate the classifier's model. The algorithm complexity is of O(n) implying a linear process.

4.2 Situation Recognition Algorithm

For the situation recognition, a similarity based search algorithm is used. Feature bundle matching approach is used, a technique recommended for temporal data (Kusiak, 2001). Similarity search is based on a feature relationship rather than feature values. For example

if featurel=X and feature2=Z then D

can be represented as

if feature 1 feature2 = XZ then D 45

Fig. 4 shows the similarity search algorithm developed.

```
function similarity ( C[1..n], P_new)
      B← feature_bundle(P_new)          /* Given new event bundle features
   count ← 0
   for i ←1 to n do                     /* for all classes exist
      B_C ← feature_bundle(C[i])        /* Bundle features in class
      if Dist(B, B_C) is 0 then         /* if features matches any class
      count ← count + 1
      match[count] = i;                 /* index classes matched
   end for
  if count > 1 then                          /*if more than one class matched
   conflict is true                     /* exist conflict
else return C[i]                        /* else copy gate decision
if conflict is true                              /* if conflict exist
   index ← refine ( C, match[1..count], resolve)      /* for each classes index
                                                 refine decision
                                                 resolve conflict
end if
if resolve true then
   return C[index];
else                                       /* if resolve is false
   probable(C, match[1...count])           /* use probable decision
   if count is 0                              /* if no match found
         add   C_m+1 ←C[m] + P_new         /*add new class
```

Figure 4: Similarity search algorithm

The algorithm is executed in linear time that increases linearly with the size of the classes hence of O(n).

5.0 RESULT

The temporal data mining technique is developed using Turbo C and tested with real data from the Timah-Tasoh dam operation in the State of Perlis, Malaysia. The dam engineering structure uses spillway gates as its flood control mechanism. Excess of water from the reservoir will be released through these gates. The data for the years 1998, 1999 and 2000 were used as training data and for the years 2001 and 2002 as testing data. Performance of the algorithm developed were measured based on parameters listed in Table 1.

5.1 Optimum Sliding Window Size

Basically the algorithm has two main parts: the *training* for the learning activity and the *testing* for the situation recognition activity. The program was executed several times to determine the optimum size of the sliding window with best performance. Each run will use different window sizes *w*. Data used is from 1998, 1999 and 2000 operations where the inputs are the daily water levels, daily rainfall measurements at six rainfall stations and daily number of gates opened. The output of this algorithm will be the number of gates predicted to be open at a particular day. Table 2 shows the results of this experiment based on the parameters described in Table I by comparing the actual and predicted decision.

Table 2: Performance with various window sizes

w	*tp*	*fp*	*tn*	*In*	*Sensitivity*	*Specificity*	*Acc*
1	2	38	327	0	1	0.8958	0.8965
2	2	18	346	0	1	0.9505	0.9508
3	1	7	356	1	0.5	0.9807	0.9781
4	1	5	357	1	0.5	0.9862	0.9835

From this experiment, the window of size two gave the best sensitivity and accuracy of result. Chart in Fig. 4 illustrates the different performance measurement for each window size where *Spec* represents the sensitivity value, *Spec* represents the specificity value and *Acc* represents the accuracy of total prediction (event and non-event).

Looking at the chart, the overall prediction performance is best when the window size is 2. It is sensitive enough to be able to detect all possible events given by the value of 1. For the prediction of non-events, there are some non-events that were predicted as events. These are considered as the false alarms. Based on Equation (1), the false alarm rate for window of size 2 is about 5%.

$$\textit{False alarm} = 1 - \textit{Specificity} \qquad (1)$$

This result implies that the sliding window used will store information for time *t-1* and *t-2*. This value indicates that the decision at the current time *t* depends on the event that happened in two preceding units in the time index. In other words, gates will be opened following rainfall patterns of the two consecutive days. Next section will discuss the results of the features extracted from the training data used for the experiment.

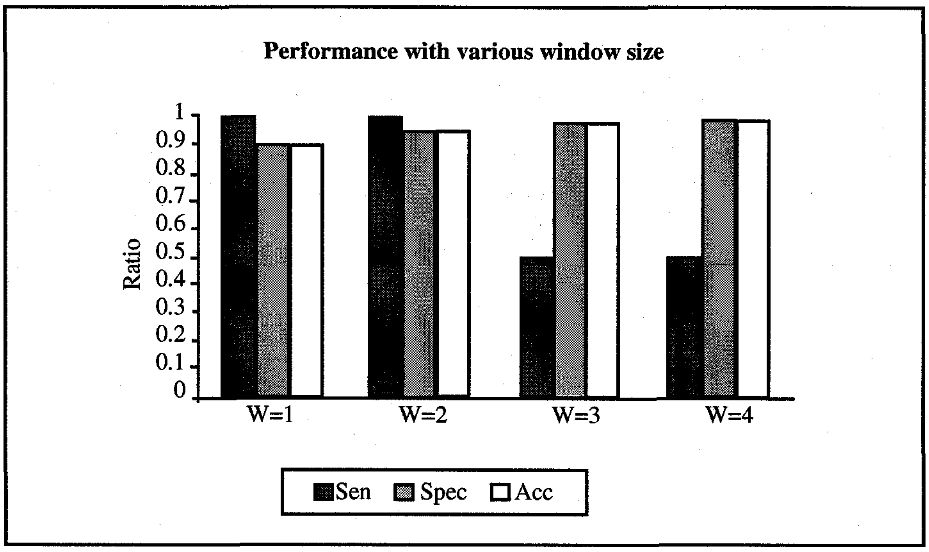

Figure 4: Comparison of prediction accuracy among various window sizes

5.2 Temporal Classifier's Model Extracted

Based on the optimum window size found, temporal classification process is executed to generate the classifier's model. The classifier represents the decision patterns related to a unique value of a decision variable. Data from 1998, 1999 and 2000 were used for training purposes. The data obtained need to be continuous and read in their temporal sequence so as to get an accurate decision pattern. The data cannot be randomly read or rearranged. This will disturb the nature of the temporal sequence. Table 3 shows the category of the parameters as determined by the DID (drainage an irrigation department) domain expert. Using the category makes the description of the situation simple to humans. For example, the statement "It is raining heavily" is easily understood compared to "It is raining 15Omm". The linguistic form is easier to the human imagination.

The original data is then transformed into a discrete representation based on the category listed in Table 3. Thus the learning algorithm extracts patterns from the data in the discrete form. Table 4 shows the temporal patterns extracted by the learning algorithm impe1emented using the training data mentioned.

Table 3: Symbolic representation of water levels and rainfall measurements according to Timah Tasoh Dam Unit, DID Perlis

Nominal Value	Water Level /m	Flood Stage	Nominal Value	Rainfall /mm	Category
0	<29.0	Normal	0	0	None
1	< 29.4	Alert	1	1-10	Light
2	< 29.6	Warning	2	11-30	Moderate
3	> 29.6	Danger	3	31-60	Heavy
			4	> 60	Very Heavy

Table 4 shows that there are two classes extracted. First, a class with two gates opened and next class with four gates opened. There are thirteen instances of patterns that lead to the opening of two gates and nine instances for the class with four gates opened. Each instance represents the pattern related to a decision. Next section will discuss on the interpretation of these patterns.

Table 4: Temporal Pattern Classes Extracted

3..1.1	**Number of Gates Open = 2**			**3..1.2**	**Number of Gates Open = 4**		
Flood Stage/t	Avg Rain t-2	t-1	%Δ	Flood Stage/t	Avg Rain t-2	t-1	%Δ
0	1	1	0.1760	0	4	4	0.7495
0	1	1	0.0701	0	2	3	0.9609
0	3	3	0.1744	0	2	1	0.0690
1	0	2	0.2226	0	3	2	0.5759
1	2	1	0.0515	0	2	2	0.3119
1	3	2	0.3607	1	2	2	0.4480
1	4	1	0.3963	1	2	2	0.2914
1	2	2	0.3442	1	2	1	0.0684
1	0	3	0.4112	1	4	2	0.3626
1	2	3	0.5659				
1	2	2	0.2059				
1	3	1	0.1711				
1	1	1	0.1025				

5.3 Experience Knowledge Representation

Referring to results in Table 4 earlier, the patterns can be translated in meaningful form that is much more readable and descriptive especially to the domain expert.

Each pattern can be transformed into a rule-based form that represents the decision rules. For example, using the description in Table 3, patterns shaded in Table 4 can be translated into:

ifjlood_stage is *Normal* at time *t*
if average rainfall is *Heavy* at time *t-2*
if average rainfall is *Heavy* at time *t-1*
then open two gates

if *flood_stage* is *Alert* at time *t*
if average rainfall is *Moderate* at time *t-2*
if average rainfall is *Heavy* at time *t-l*
then open two gates

if *flood_stage* is *Alert* at time *t*
if average rainfall is V*ery Heavy* at time *t-2*
if average rainfall is *Moderate* at time *t-I*
then open four gates

This rule-based form can be used for designing a knowledge based system to support emergency decision-making, is in a readable form and easy to understand.

5.4 Performance of Temporal Data Mining Algorithm

In the experiment, two approaches of feature bundle matching were used. First approach, bundles only the rainfall features. The second approach bundles the flood stage and rainfall patterns together as shown in Table 5 below.

Table 5: Two approaches of feature bundle matching

Approach 1	**Approach 2**
If flood stage = X at tme *t* if *rainfall* at *t-I*, *t-2* = *YZ* then?	*If flood stage* at time *t*, *rainfall* at *t-1*,*t-2* = *XYZ* then?

Table 6 shows the comparison of performance results for each approach according to the data used for testing.

Table 6: Performance of temporal data mining algorithm

Dataset	*tp*	*fp*	*tn*	*fn*	*Sen*	*Spec*	*Acc*
1998							
Approach 1	2	12	272	0	1	0.958	0.958
Approach 2	2	32	252	0	1	0.8873	0.8810
1999							
Approach 1	9	21	334	0	1	0.941	0.942
Approach 2	8	23	333	1	0.8751	0.9354	0.9341
2000							
Approach 1	11	18	342	0	1	0.950	0.9164
Approach 2	9	31	331	0	1	0.9144	0.95
2001							
Approach 1	2	12	351	0	1	0.967	0.967
Approach 2	0	14	350	1	0	0.9615	0.9589
2002							
Approach 1	0	4	145	0	1	0.973	0.973
Approach 2	0	2	147	0	1	0.9866	0.9866

Chart in Figure 5 shows the companson tor approach 1.

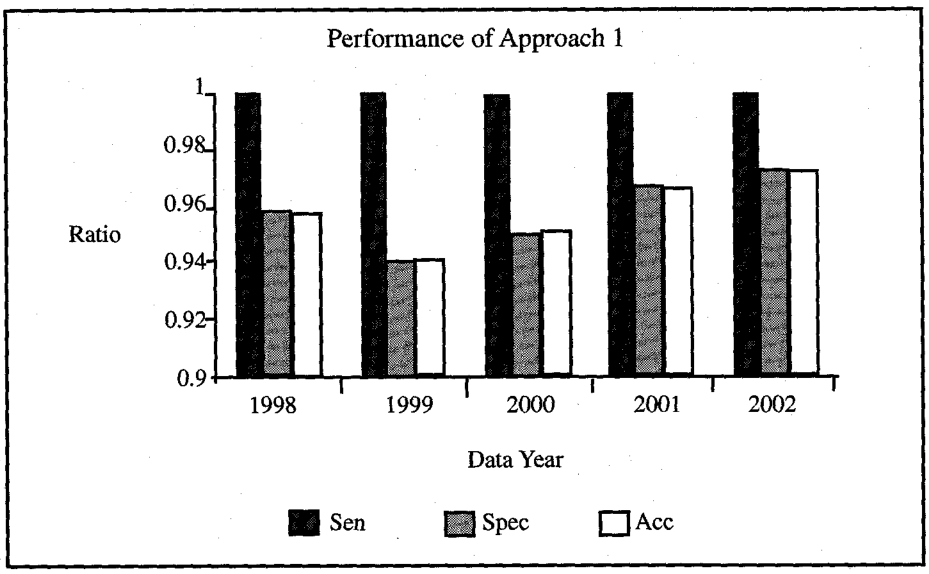

Figure 5: Performance of temporal data mining algorithm using approach 1

Approach 1 has a good sensitivity in prediction shown by the value of 1 for all the data tested. The rate of false alarm is about 3% to 6% with an accuracy of prediction above 94%. Chart in Fig. 6 shows the performance using approach 2.

Based on chart in Fig. 6, the approach is not sensitive to one of the test data that is 1999 with the value of 87.5%. The rate of false alarm is also higher in this approach, which is about 1 % to 11 %. The prediction accuracy is above 88%. The algorithm does not perform well in the earlier data. From these two results, it is concluded that approach 1 is a better strategy for the feature bundle. This approach uses rainfall patterns bundled together. This result is in agreement with Teh and Ong (2000) recommendation of using rainfall as the triggering mechanism for flood warning. Overall the algorithm is able to predict the opening of the spillway gate with more than 90% accuracy with less than 10% false alarm.

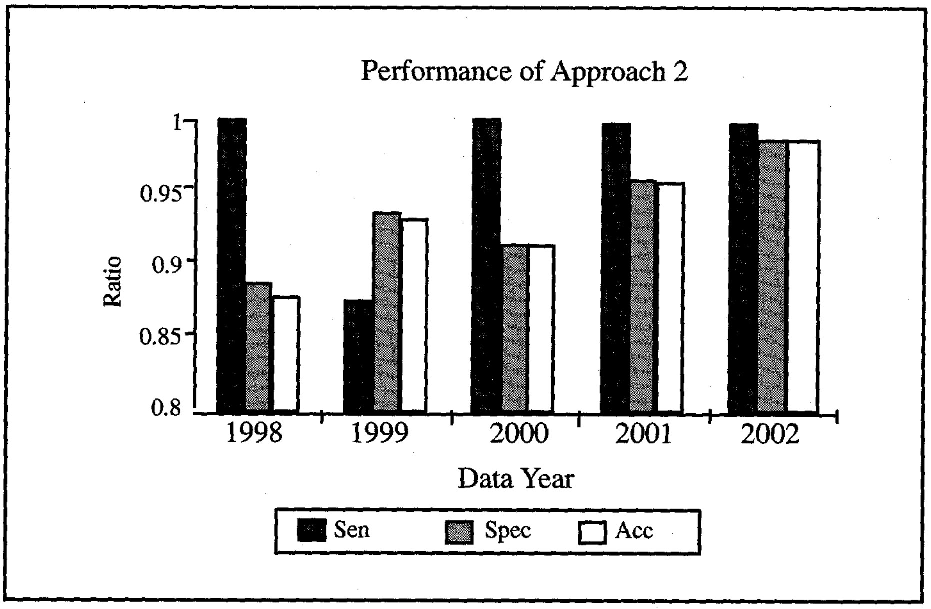

Figure 6: Performance of temporal data mining algorithm using approach 2

6.0 CONCLUSION

This study demonstrated the technical feasibility of adopting temporal data mining approach for representing the decision process of RPD model. By using previous decision patterns a more rapid decision can be reached rather than plan from scratch. Real reservoir operation data was used to evaluate the algorithm performance. Based on the results, the optimum sliding window size is two. This implies an event can be predicted by the two preceding days event. The patterns were extracted from the training data that can be transformed into a readable and descriptive rule-based form. A new set of data was used to test the classifier. The overall performance of the temporal data mining algorithm in predicting spillway gate operations is above 90% with less than 10% false alarm. From the experiment, the feature bundle approach shows that rainfall is a better triggering factor for predicting floods where the performance of Approach has a better accuracy. For future work, integrating a fuzzy logic component in the feature extraction could improve the quality of the patterns extracted hence the prediction rules.

7.0 REFERENCES

Brann, D. M., Thurman, D.A. & Mitchell, C. M. (1995). Case-based Reasoning as a Methodology for Accumulating Human Expertise for Discrete System Control. In *Proc. of the* 1995 *IEEE International Confrenee on Systems, Man,* & *Cybernetics.* Vancouver, Canada.

Breen, B. (2000). What's Your Intuition. *The Magazine.* [Online]. Retrieved 15 Jan 2002 from http://fascompany.com/online/38/klein.html

Craig, P. A. (1998). Improving pilot decision-making situations of high stakes, high stress, & time pressure. PhD Thesis. Tennessee State University.

Currey, J. & Botti, M. (2003). Naturalistic Decision-Making: A Model to Overcome Methodological Challenges in the Study of Critical Care Nurses' Decision-Making About Patients' Hemodynamic Status. *American Journal of Critical care.*

Deitch, E. L. (2001). *Learning to Land: A Qualitative Examination of Preflight* & *in Flight Decision-making Process in Expert* & *Novice Aviators.* PhD thesis. Faculty of Virginia Polytechnic Institute & State University

Dunham, M. H. (2003). *Data Mining: Introductory and Advanced Topics.* New Jersey: Prentice Hall.

Frawley, W., Piatetsky-Shapiro, G. & Matheus, C. (1992). Knowledge Discovery in Databases: An Overview. *AI Magazine.*

Grzymala-Busse, J. W., Zheng, X., Goodwin, L. K. & Grzymala-Busse, W.J. (1999). Comparing Different Strategies in Using Rules Induced from Pre-term Birth Data. *In Proc of the Information Processing, & Management of Uncertainty in Knowledge Based System* (Madrid, Spain.)

Karimi, K. & Hamilton, H. J. (2002). Discovering Temporal Rules from Temporally Ordered Data. In H. Ym, N. Allinson, R. Freeman, J. Keane, S. Hubbard(eds). *Lee. Notes on Computer Science* Vol 2412 , Springer.

Keogh, E. (2001). Tutorial on Indexing & Mining Time Series Data. ICDM '01. The *2001 IEEE International Conference on Data Mining,* Nov. 29, San Jose.

Klein, G. & Klinger, D. (2002). Naturalistic Decision-Making. *Human Systems* IAC Gateway.

Klein, G. & Weick, K. E. (2000). Decisions: Making the Right Ones. Learning from the Wrong Ones. *Across the Board, June 2000.* Retrieved 10 January 2002 from http://www.conference-board.org/atb/artic1e/ kleinJun.cfm,2000

Kusiak, A. (2001). Feature Transformation Methods in Data Mining. *IEEE Transactions on Electronics Packaging Manufacturing.*

Kusiak, A. (2002). Data Mining & Decision-Making. In B.V. Dasarathy (ed).
Proceedings Of the SPIE onference on Data Mining & *Knowledge Discovery: Theory, Tools,* & *Technology IV* 4730 . Orlando, Florida. 155-165.

Kusiak, A., Kern, J. A., Kernstine, K. H. & Tseng, B. T. L. (2000). Autonomous Decision-making: A Data mining approach. *IEEE Transactions on Information Technology in BioMedicine.*

Liang, Y., Robichaud, F. & Fugere, B. J. (2001). Implementing a Naturalistic Command Agent Design. *In Proc. of the* 1 *O" Computer Generated Forces*, Norfolk, VA. 15-17 May.

Lipshitz, R., Klein, G. & Orasanu, J. (2000). *Taking Stock of Naturalistic Decision-making.* Center for Study of Organization & Human Resource Management. Retrieved 20 January, 2002, from http:// organizations.haifa.ac.il/raanaan-taking.doc

Morris, J. G. & Mitchell, C. M. (1995). A Designer Associate for Command and Control Software Development. In *Proc of the* 1995 *IEEE International Conference on Systems, Man* & *Cybernetics,* Oct. Vancouver, BC.

Norling, E., Sonenber, L. & Ronnquist, R. (2000). Enhancing Multiagent Based Simulation with Human-like Decision-making Strategies. In S. Moss, & P. Davidson (eds) *Multiagent based Simulation: Proceedings of the 2nd International Workshop MABS2000.* Springer, Boston.

Norwawi, N. M. (2004). *Computational Recognition-Primed Decision Model based on Temporal Data Mining Approach in a Multiagent Environment for Reservoir Flood Control Decision.* PhD Thesis. Universiti Utara Malaysia.

O'Shaughnessy, J. (1999). Sources of Power: How People Make Decisions. [Review ofthe Book]. *J. of Macromarketing.* June.

Povinnelli, R. J. & Feng, X. (2002). A New Temporal Pattern Identification Method for Characterization & Prediction of Complex Time Series Events. *IEEE Transactions on Knowledge,* & *Data Engineering.* (in press)

Robichaud, F. (2001). *Implementing a Naturalistic Command Agent Design.*
Paper presented at the Workshop on Computerized Representation of RPD. Oct. Boulder, CO.

Teh, S. K. &Ong, S. H. (2000). *Strategy for Flood Response.* Persidangan Tahunan Pengurus Kanan JPS Malaysia 2000, 7-8 Sep. Langkawi, Kedah.

Warwick, W., McIlwaine, S., Hutton, R. & Dermott, P. (2001). Developing Computational Models of Recognition-primed Decision-making. In *Proceedings of the 1 O" Conf. On Computer Generated Forces,* Norfolk. VA. 15-17 May.

Wong, B. L. W. (2000). The Integrated Decision Model in Emergency Dispatch Management & its Implication for Design. *The Australian Journal of lnformation System(AJ1S). 7(2).*