行政院國家科學委員會專題研究計畫　成果報告

混合式柔性計算系統於營建知識發掘之研究(I)

計畫類別：個別型計畫
計畫編號：NSC92-2211-E-216-017-
執行期間：92年08月01日至93年07月31日
執行單位：中華大學營建工程學系

計畫主持人：余文德

計畫參與人員：林涵文

報告類型：精簡報告

處理方式：本計畫可公開查詢

中　華　民　國 93年10月8日

行政院國家科學委員會補助專題研究計畫成果報告

※※※※※※※※※※※※※※※※※※※※※※※※※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※　混合式柔性計算系統於營建知識發掘之研究(I)　※
※　　　　　　　　　　　　　　　　　　　　　　　　※
※※※※※※※※※※※※※※※※※※※※※※※※※

計畫類別：■個別型計畫　　□整合型計畫

計畫編號：NSC 92-2211-E-216-017

執行期間：　92 年　8 月　1 日至　93 年　7 月　31 日

計畫主持人：余文德

共同主持人：

本成果報告包括以下應繳交之附件：

□赴國外出差或研習心得報告一份
□赴大陸地區出差或研習心得報告一份
□出席國際學術會議心得報告及發表之論文各一份
□國際合作研究計畫國外研究報告書一份

執行單位：中華大學營建工程學系

中　華　民　國　93 年　10 月　8 日

行政院國家科學委員會專題研究計畫成果報告

國科會專題研究計畫成果報告撰寫格式說明

Preparation of NSC Project Reports

計畫編號： NSC 92-2211-E-216-017

執行期限：92 年 8 月 1 日至 93 年 7 月 31 日

主持人：余文德　中華大學營建工程學系

計畫參與人員：林涵文　中華大學營建管理研究所

一、中文摘要

營建工程為一以經驗為基礎之專業，從過去執行專案中累積與發掘之營建工程知識為各營建業者(包括業主、設計者、施工單位及營建管理顧問)確保未來專案成功之基石。因此營建業也屬於一種特殊的「知識經濟」產業。然而專業知識之累積不易，因此如何從過去所完成的歷史案例資料中去發掘有用之資訊，並轉換為工程師與管理者能夠應用的實務知識，更是一件關鍵的任務。近年來由於知識發掘(KDD)與資料探勘(DM)技術的突飛猛進，使得其他產業在歷史資料加值處理與創造知識商品上得到極大進展。然而營建業在此一領域的應用卻極為有限，究其原因不外兩個主要理由：(1) 營建業對於知識發掘與資料探勘技術感到陌生而不知採用；(2) 現有的知識發掘與資料探勘技術對於營建業的特殊資料性質無法有效應用而被捨棄。然而我國營建業若欲擺脫目前之困境，朝向知識經濟轉型，則上述原障礙必須加以排除。本研究旨在探討營建業在知識發掘上所遭遇之問題，主要著眼點有二：(1) 發展出適用於處理營建特殊資料性質之資料探勘演算法；(2) 發掘出具有「可讀性」之營建工程知識，俾知識之傳承、再利用與專家驗證。為達到此一目標，本研究將廣泛地回顧現有之知識發掘與資料探勘方法，並分析營建業在應用此一技術時所遭遇之特殊問題及障礙。其次詳細回顧並分析目前資料探勘技術中最常用之各種柔性計算(Soft computing)方法(包括模糊理論、類神經網路、基因演算法、粗集合理論及案例式推理等)在營建業知識發掘應用之適用性，並針對營建業特性提出一資料探勘演算法，以處理營建資料不確定、不完整、不純及數量不足等問題。從現有的文獻得知，上述五種理論在資料探勘用各有其長處。其中模糊理論在表達專家知識以及處理不確定之資訊上已成為普遍接受的方法；類神經網路則在分佈型知識庫、平行處理及局部最佳化搜尋方面之應用極為成功；基因演算法在處理非結構性映至問題及全域搜索有強大之功能；粗集合理論在分析缺漏及不完整之資料，並用以萃取決策法則上之應用方面展現其卓越潛力；案例式推理方法在資料量極少的情況下即可提供相當有用之推理結果。本研究整合柔性計算理論以發展出一套可應用於營建資料庫之「工程知識發掘系統」的演算法，並依據所推導出之演算法撰寫一套電腦軟體程式，以驗證所建構之演算法的可行性與正確性。第一年之研究重點在於建立類神經模糊柔性系統之計算架構，並針對資料缺漏問題進行探勘。

關鍵詞：資料探勘、知識發掘、柔性計算、類神經模糊系統

Abstract

Construction has been conceived as an experience-based discipline. Knowledge learned from previous projects plays important role in successful performance of future projects. This has made construction an ideal industry for knowledge based economy. However, modern KDD (knowledge discovery in databases) or DM (data mining) technologies are not yet widely exploited and adopted in the field of construction engineering and management. This is due to two main causes: (1) the construction industry is not familiar with KDD and DM technologies; (2) the available KDD and DM technologies do not fit the special characteristics of data in the field of construction engineering and management. Should the construction industry be pursuing knowledge based economy, obstacles caused by the above two reasons must be removed and the reusable domain knowledge must be generated from historical data. For this end, this research is proposed to tackle problems encountered in knowledge discovery in real world construction databases. The focuses are: (1) development of DM algorithms for the knowledge discovery of the unique construction data characteristics; (2) generation of human understandable knowledge, so that domain experts can visualize and verify it. At first, the existing KDD and DM methods are reviewed. Problems faced in applications of KDD and DM for construction engineering and management are broadly surveyed to identify the special characteristics of construction data, which hinder the implementation of KDD and DM in construction industry. The existing soft computing techniques, including fuzzy sets, artificial neural networks, genetic algorithms, rough sets, and case-base reasoning, are thoroughly reviewed to propose the most appropriate hybridization for handling unique domain data characteristics. The data mining algorithms are developed to discover knowledge from construction data, which are usually uncertain, incomplete, partial true, and scarce in their nature. A Hybrid Soft Computing System has been developed for implementation of data mining and knowledge discovery in construction industry. The main work of the first year was on developing the architecture of a neuro-fuzzy system with focus on mining incomplete databases.

Keywords: Data mining, knowledge discovery in databases, soft computing, neuro-fuzzy systems

二、緣由與目的

The research of this year aimed at developing a neuro-fuzzy system (NFS), based on the original Fuzzy Adaptive Learning Control Network (FALCON) model proposed by Lin and Lee [2], which is modified and improved so that it is able to handle historical data with partial-missing attribute values. As the proposed method is based on NFS architecture, it is equipped with both learning and reasoning capabilities and is able to mine construction knowledge from historical data. It also provides explanation of the reasoning process for system users to develop improvement strategies. A variable-attribute NFS network along with the associated learning algorithms is developed. The proposed variable-attribute NFS network is very useful not only for data mining but also for real-time decision making when the complete information cannot be acquired or when it is too expensive to collect.

三、問題定義

Incomplete data are omnipresent in the traditional construction databases due to the harsh outdoor environment, the attitudes of workmen who collect the data, and merging of different databases. Two categories of incompleteness are defined as follows: (1) missing data—incomplete coverage of data in some intervals of the universe of discourse; (2) missing values—incomplete information in some interesting attributes of a dataset. Following describes the problems of the two types of data incompleteness.

3.1 Missing data

The first type of data incompleteness problem is lack of data sets in some intervals of the universe of discourse for a specific attribute. This type of data incompleteness can be further classified into two categories: (1) interpolation type; (2) extrapolation type. For the interpolation problems, the datasets are missing between two clusters of data in the universe of discourse of an interested attribute (see Figure 1). Thus, the missing data are usually recovered by interpolation schemes.

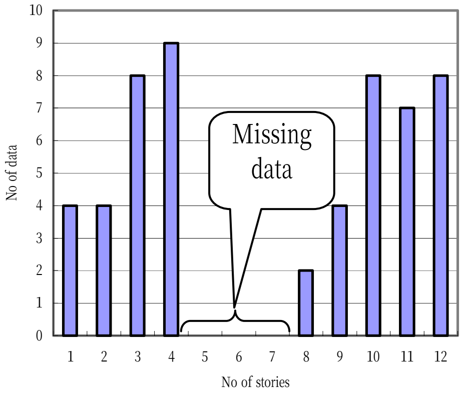


Figure 1. Interpolation type of missing data

On the other hand, for the extrapolation problems, the data sets are missing at extremes parts of the universe of discourse of an interested attribute. Thus, extrapolation schemes are adopted for data recovery. Figure 2 shows an example of extrapolation type of missing data.

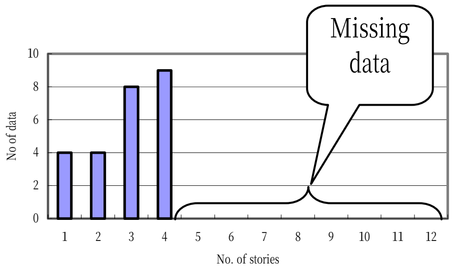


Figure 2. Extrapolation type of missing data

3.2 Missing values

In the second type of data incompleteness problem, some values of interesting attributes may not be available due to numbers of reasons: (1) the attributes are not considered important at time of data entry; (2) the mistakes made by collector; and (3) equipment malfunctions. This type of problem is more severe while merging databases from different sources. For example, in Table 1, data sources of firm A, B, and C provide inconsistent data format. The resulted database shows a typical example of missing-value type of data incompleteness, where the missing attribute values are depicted with shadowed cells.

Table1 Missing values in heterogeneous databases

Firm	weath.	Temp.	Humid.	item	Prod.
A	cloud	18			high
B		23	90%		low
C			60%	Form.	Ave.

3.3 Definitions of data incompleteness in this research

This research tackles problems regarding to "missing values" rather than "missing data". That is, the data incompleteness is defined as the percentage of unavailable attribute values. The missing data problem is not considered in this research. In order to evaluate the degrees of incompleteness of missing values, two measures of data incompleteness are defined: (1) percentage of incomplete attributes (*PIA*)—measuring the ratio of the number of unavailable attributes (which consist at least one missing value) over the number of total attributes, e.g., the *PIA* of the 3 datasets in Table 1 is $\frac{4}{4}=100\%$; (2) percentage of incomplete datasets (*PID*)—measuring the ratio of the number of datasets with at least one missing value over the number of total datasets, e.g., the *PID* of the 3 datasets in Table 1 is $\frac{3}{3}=100\%$; (3) percentage of overall incompleteness (*POI*)—measuring the ratio of the total number of incomplete attributes over the number of all attribute values of all datasets, e.g., the *POI* of the 3 datasets in Table 1 is $\frac{6}{3\times 4}=50\%$. The *PIA* is measured because that in the traditional data cleaning method (will be discussed in the next section) the tuples with incomplete attribute information will be discarded or processed manually. Therefore, all of the four tuples may be discarded by data cleaning. Similarly, the *PID* measures how many datasets will be discarded or processed in traditional data cleaning process. The *POI* is an overall assessment of data incompleteness Note that, in calculation of the above three measures of data incompleteness, only the precondition attributes characterizing a data set are considered, the consequence part of a data set, such as the last column (productivity) in Table 1, is not included.

四、研究方法

This research proposes a modified FALCON, namely Variable-Attribute Fuzzy Adaptive Logic Control Network (VaFALCON), for mining of incomplete construction data. The data incompleteness is defined previously as *PIA* (percentage of incomplete attributes), *PID* (percentage of incomplete datasets), and *POI* (percentage of overall incompleteness). In order to improve the drawbacks of traditional data cleaning methods, the proposed VaFALCON aims at handling incomplete construction data without restrictions on *PIA*, *PID*, or *POI*. That is, it is expected to handle the data incompleteness at any degrees of severity. Moreover, the proposed VaFALCON is designed to take incomplete construction data directly without data cleaning. It does not mean that the data cleaning process is not recommended before DM, proper and correct pre-processing of dirty data will always help the performance of data mining. However for VaFALCON, any pretreatment on filling up of missing data or aggregating of data is not required. Such relaxation of data pre-processing requirement provides a chance to preserve the originality of the raw data as much as possible. Sometimes data cleaning may cause loss of original information of data as discussed previously in Section 4.

Following describes the details of the proposed VaFALCON.

4.1 Variable-attribute network structure (VANS)

Most of the current data mining techniques, including traditional FALCON and artificial neural networks, take only complete data without missing values as their inputs. If the input data are incomplete, data cleaning is performed to fill out the missing values so that the algorithms of DM techniques can work. The core idea of VaFALCON is based on such concept. So, the first step for developing VaFALCON is to establish a mechanism for processing variable numbers of attributes. Such mechanism is called "variable-attribute network structure (VANS)".

Using the example FALCON model shown in Figure 5 as a basis for discussion, there are three input attributes (X_1, X_2, and X_3) and one single output (Y). Each of the three input attributes is fuzzified into two fuzzy linguistic terms. The output is fuzzified into four fuzzy linguistic terms. Referring to Table 2, data set *A* is a complete data that consist of input and output pairs as ([a,b,c], D), where [a,b,c] is the vector of inputs and D is the value of output. The other incomplete data set $\bar{A}$ contains a missing attribute X_1 whose value is unknown and denoted as "*nan*" meaning "not a number".

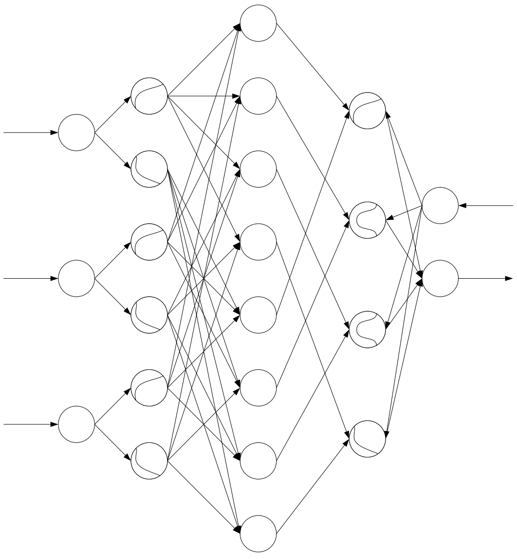


Figure 5. Example FALCON model

Table 2 Complete vs. incomplete data set

Attribute	X_1	X_2	X_3	Y
Data set *A*	a	b	c	D
Data set $\bar{A}$	*nan*[*]	b	c	D

[*]Not a number (empty)

The propagation of a complete data set in FALCON is shown in Figure 6, where the interconnections between the first two nodes in Layer 2 (pre-condition fuzzy linguistic terms) and Layer 3 (rule nodes) are shown as solid links, which means physical connections.

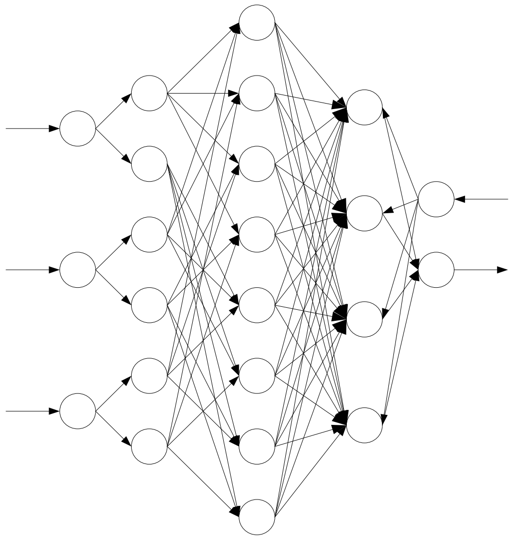
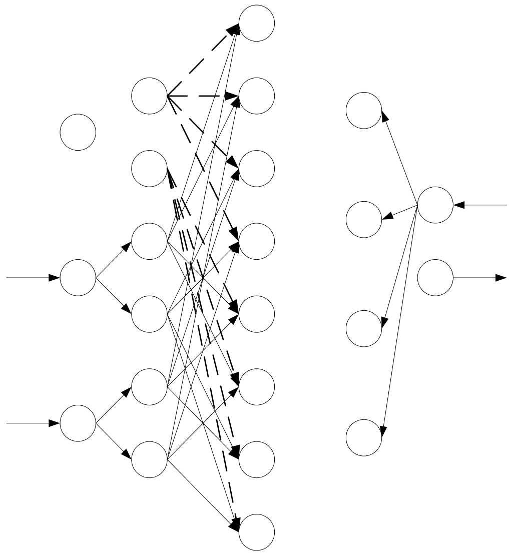


Figure 6. Connections of FALCON for complete data

As the first input (X_1) is missing, the resulted FALCON is shown in Figure 7. In Figure 7, the fuzzy linguistic term nodes of the first input (X_1) are disconnected (shown as dashed line) with the rule nodes in the following layer. The signals are not propagated via dashed-line links. In the traditional FALCON, the network of Figure 7 is not trainable due to the undetermined links between Layer 3 and Layer 4, so that the signals cannot be propagated.

a



b

c

Figure 7. Connections of FALCON for incomplete data

Using the first principle of data cleaning described in Section 4, the first tuple (X_1) can be ignored while the information of the other two tuples (X_2 and X_3) are conserved. To make the best use of the data information, the FALCON is degraded to the one shown in Figure 8, where the first input is omitted. With the two residual tuples, the FALCON of Figure 8 is trainable as the signal propagation path is determined.

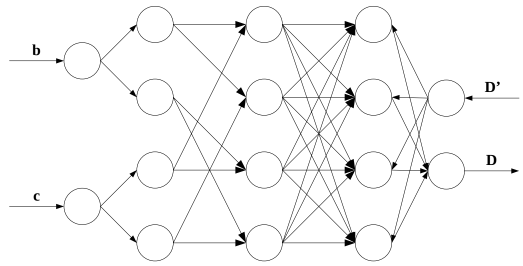


Figure 8. Degraded FALCON for incomplete data

By the process of network degradation with missing attribute of VANS, the structure of FALCON is variable according to the training data. The VANS enables FALCON to handle datasets with any degree of data incompleteness.

4.2 Modified learning algorithms

In order to implement the VANS of VaFALCON described previously, the learning algorithms of original FALCON are modified as following:

(1) Modification of Kohonen learning rule [11]
In order to take incomplete input data, Kohonen learning rule is modified as shown in equation (7) for learning the means of membership functions.

If $x \neq nan$

$\hat{w}_i^{k+1} = \hat{w}_i^k + \eta^k \left(x - \hat{w}_i^k \right)$

$\hat{w}_j^k = \hat{w}_j^k$, for $j = 1, 2, \ldots, n \quad j \neq i$.

(7)

end

(2) Modification of first-nearest-neighbor heuristic
In order to determine the primitive spreads of membership functions, the first-nearest-neighbor heuristic is modified as shown in equation (8).

```
if $x \neq nan$
    $\sigma_i = \frac{|m_i - m_{nearest}|}{\gamma}$
    (8)
end
```

In both equation (1) and (2), "*nan*" (not a number) means missing of attribute value. The logic judgment in the first line of equation (1) and (2) represents that the modification is performed only when the attribute is not empty.

(3) Modification of *fuzzy AND* inference

The *fuzzy AND* inference of rule nodes in the third layer of FALCON performs t-norm computation, i.e., minimization or intersection. In VaFALCON, it should be avoided to take the memberships of missing attributes as the outputs (i.e., minima) of *fuzzy AND* operations. To achieve this goal, the memberships of missing attributes are replaced with a constant greater than 1.0 (the *maximum* value of membership function), so that it won't become the *minimum* of any *fuzzy AND* operation. Here "1.1" is adopted as shown in Table 3 and Table 4. After the modifications described above, it is guaranteed that no missing attribute will become the output of fuzzy AND operations.

Table 3 Original outputs of Layer 2 in VaFALCON

Node	1	2	3	4	5	6
Output	*nan*	*nan*	0.825	0.236	0.148	0.567

Table 4 Modified input of Layer 3 in VaFALCON

Node	1	2	3	4	5	6
Output	1.1	1.1	0.825	0.236	0.148	0.567

4.3 Learning process of VaFALCON

The learning process of VaFALCON is similar to the original FALCON except that detailed computations are recoded to implement the above mentioned modifications. The learning process consists of two phases: (1) Self-organizing phase—including modified Kohonen learning and reinforcement competitive learning to construct the primitive FALCON structure; (2) Back-propagation phase—applying back-propagation learning rule to fine-tune the network. The learning process ends when the expected error rate is achieved or the maximum number of learning cycles is reached.

五、研究結果

A VaFALCON neuro fuzzy system is programmed with Matlab™ v6.3 to perform functions proposed in the previous section. The developed VaFALCON is then tested on a PC platform with Pentium IV 1.5G CPU, 1.0 GB SRAM, and Windows 2000® operating system. Following describes the details of the testing experiments.

5.1 Description of testing experiments

In order to verify the proposed VaFALCON, three example construction databases collected from published literature are selected for system testing of the proposed VaFALCON including: (1) building construction cost estimation selected from [12]; (2) structural cost estimation selected from [13]; and (3) estimation of the curtain wall construction duration selected from [[2]]. All of the three examples are scarce (limited in numbers of data), it is therefore not affordable to discard any incomplete dataset. The best policy is to make the best use of the available incomplete data.

Three scenarios of experiments are tested with VaFALCON for the above three examples: (1) learning of complete data set; (2) learning of incomplete data discarding the data sets with missing values by traditional data cleaning method; (3) learning of data sets with various degrees of data incompleteness in terms of *PIA*, *PID*, and *POI*. The result of the first scenario is used as basis for comparison.

The extreme conditions are simulated for data incompleteness by setting the *PIA* and *PID* close to 100%. That is, at least one missing value is found in almost every dataset and every tuple. Such extreme conditions are very difficult to process with the traditional data cleaning methods described in Section 4. The *POI* is varied from 5% to 25% (or higher), resulted arbitrarily by the random selection process. The incomplete datasets are generated by a random selection process that picks the locations of missing attributes randomly. At first, the attribute locations are numbered sequentially on the table of the data. Then, a random number is generated and timed by the total number of data attributes. This process is repeated until target number of missing attribute values is reached. The testing accuracy is defined by the following equation:

$$Acc.(\%) = \left\{1 - Abs\left(1 - \frac{Estimated}{Actual}\right)\right\} \times 100\%,\text{(9)}$$

where *Estimated* is the output generated by the system, *Actual* is the actual result observed from real world, and *Acc.* is the percentage accuracy of the estimation. The absolute value is taken within the parenthesis to avoid minus values.

5.2 Example I—building construction cost database

(1) Data preparation

The first example is selected from a real world construction database published in Yu [12]. Building construction cost estimation is a difficult task during the early stage of a construction project as most design information is not available at that stage. Traditional approaches rely on domain experts (experienced cost estimators) in performing the conceptual cost estimation. However, the domain experts are difficult to find, expensive to educate, and likely to leave. Recently, the AI approaches have been widely applied in building construction cost estimations. The AI techniques are usually combined with parametric estimation methods to establish the relationships between the parametric attributes and the estimation results. The estimation quality of such methods depends heavily on the quality of the historical data. Unfortunately, the missing of attribute values is commonly found in historical cost databases due to reasons described in Section 3.

In the selected example, 4 attributes were identified as attributes among the nearly 30 parameters originally collected, including (1) earth retaining method; (2) No. of floors above ground; (3) No. of floors under ground; (4) total floor area. One single output, construction cost estimation, is recorded in the database. Totally 25 data are collected from historical building construction project by surveying the final project reports provided by public owners. 22 data sets are used for learning and the rest 3 data are used for testing. The data are shown in Table A.1. Notice that the values of the first attribute (earth retaining method) have been transformed from symbolic data into numeric data by: 1means steel rail pile, 2 means replace aggregate method, and 3 means curtain wall method. Since the order of data in Table A.1 has been randomized from their original sequence, the last 3 (shaded) datasets of the 25 data in Table A.1 are selected as testing sets. The rest are used for system training. The data incompleteness is simulated by random selections process as described previously. For the first example, five various degrees of data incompleteness are simulated as shown in Table 5.

Table 5. Data incompleteness cases of Ex. I

Measures	Case				
	(1)	(2)	(3)	(4)	(5)
PIA (%)	75	75	100	100	100
PID (%)	18	32	50	50	100
POI (%)	5	9	14	18	25

It is observed from Table 5 that the data incompleteness measures are increasing as the case numbering increases. That is Case (1) is the slightest among the five cases in data incompleteness, while Case (5) is the severest. In Case (5), both *PIA* and *PID* are 100%. It means that every tuple and dataset consists at least one missing attribute value, i.e., all dataset are dirty. The *POI* of 25% in Case (5) means that one quarter of the total attribute values in the database are missing. It can be considered as very severe data incompleteness case.

(2) Testing results

There are three scenarios to be performed for Example I. The first scenario is performed with the complete 22 datasets. The testing sets are kept complete for all scenarios to control the influential factors. The results of the three testing data are shown in Table 6. It is found that the average system accuracy is 94.66% for training sets and 92.63% for the three testing sets.

Table 6. Testing result of complete data—Ex. I

Data		Accuracy
Training sets		94.66%
Testing sets	Dataset A	88.37%
	Dataset B	91.99%
	Dataset C	97.53%
	Average	92.63%

In Scenario (2), the incomplete data are tested with FALCON by discarding the datasets with missing attribute values. The numbers of data in training and testing sets for each case are shown in Table 7. The number of available training sets is relevant to *PID*. It is noticed in Table 7 that the training set of Case (5) is empty since all data are dirty and discarded after cleaning. Thus, the testing of Case (5) is omitted.

Table 7. No. of training sets for the cases of Ex. I

	Case				
	(1)	(2)	(3)	(4)	(5)
PID (%)	18	32	50	50	100
No. of training sets	18	15	11	11	0
No. of testing sets	3	3	3	3	-

On the other hand, Scenario (3) is tested with different degrees of incompleteness as shown in Table 5 by direct learning on incomplete data. The average accuracy of the three testing sets for the fives cases in Scenario (2) and (3) are shown in Table 8.

It is found from Table 8 that the proposed VaFALCON, Scenario (3), improves the system accuracy significantly by learning the incomplete data directly compared with discarding the incompleteness data in Scenario (2). While comparing with complete data, the proposed VaFALCON can recover the system accuracy from 81%, for Case (5), up to 98%, for Case (1).

Table 8. Testing results of Ex. I

		System accuracy (%)				
Case		(1)	(2)	(3)	(4)	(5)
Scen a-rio	(1)	92.7	92.7	92.7	92.7	92.7
	(2)	75.7	62.2	59.0	38.5	-
	(3)	90.7	89.6	86.5	83.4	74.7
Difference *		15.0	27.4	27.5	44.9	74.7
Recovery**		98%	97%	93%	90%	81%

*The difference of accuracy (%) between Scenario (2) and (3)

**Recovery of accuracy: Acc. of Scenario (2)/Acc. of Scenario (3)

5.3 Example II—structural cost estimation database

(1) Data preparation

The second example is tested with a structural cost estimation database similar to the first example, however selected from a different published literature by Hsieh [13]. Totally 22 examples are collected. Among which, 20 datasets are randomly selected for training and the rest 2 datasets are used for testing. Similar to the first example, four input attributes are identified: (1) total floor area; (2) area of exterior wall; (3) No. of floors above ground; (4) No. of floors under ground. The single output is the unit cost of structural construction.

The data are shown in Table A.2. Since all attribute values are numeric, no transformation is required. Similar to Example I, the order of data in Table A.2 has been randomized from their original sequence. The last 2 (shaded) datasets of the 22 data in Table A.2 are selected as testing sets. The rest 20 datasets are used for system training. The data incompleteness is simulated by random selections as described in Example I. There are five cases with various degrees of data incompleteness simulated as shown in Table 9.

Table 9. Data incompleteness cases of Ex. II

Measures	Case				
	(1)	(2)	(3)	(4)	(5)
PIA (%)	75	75	100	100	100
PID (%)	15	30	50	60	100
POI (%)	5	10	15	20	28

Similar to Example I, the data incompleteness measures are increasing as the case numbering increases. The *POI* of Case (5) is 28%, while it is only 5% in Case (10). Both *PIA* and *PID* are 100% for Case (5), therefore no data is left after data cleaning of this case.

(2) Testing results

For the first scenario, the DM is performed on 20 complete datasets. The testing results of the two testing data are shown in Table 10. It is found that the average system accuracy is 89.31% for training sets, while it's unusually much higher (96.79%) for the two testing sets.

Table 10. Testing result of complete data—Ex. II

Data		Accuracy
Training sets		89.31%
Testing sets	Dataset A	96.30%
	Dataset B	97.28%
	Average	96.79%

Similar to Example I, the incomplete data are tested with FALCON by discarding the datasets with missing attribute values for Scenario (2). The numbers of data in training and testing sets for each case are shown in Table 11.

Table 11. No. of training sets for the cases of Ex. II

	Case				
	(1)	(2)	(3)	(4)	(5)
PID (%)	18	32	50	50	100
No. of training sets	17	14	10	8	0
No. of testing sets	2	2	2	2	-

On the other hand, Scenario (3) is tested with different degrees of incompleteness as shown in Table 9 by direct learning on incomplete data. The average accuracy of the three testing sets for the fives cases in Scenario (2) and (3) are shown in Table 12.

It is found from Table 12 that the proposed VaFALCON, Scenario (3), improves the system accuracy significantly by learning the incomplete data directly compared with Scenario (2). While comparing with complete data, the proposed VaFALCON can recover the system accuracy from 87%, for Case (5), up to 99%, for Case (1).

Table 12. Testing results of Ex. II

		System accuracy (%)				
Case		(1)	(2)	(3)	(4)	(5)
Scena-rio	(1)	96.8	96.8	96.8	96.8	96.8
	(2)	84.5	81.3	74.0	72.9	-
	(3)	95.4	91.8	90.5	89.8	84.5
Difference*		10.9	10.5	16.5	16.9	84.5
Recovery**		99%	95%	93%	93%	87%

*The difference of accuracy (%) between Scenario (2) and (3)
**Recovery of accuracy: Acc. of Scenario (2)/Acc. of Scenario (3)

5.4 Example III—curtain wall construction duration estimation database

(1) Data preparation

In the third example, data of the construction duration of under ground curtain wall are collected from a published literature by Yang [[2]]. Since curtain wall method has been widely adopted in urban construction projects. Social costs can be very high under inappropriate management practice. Therefore, the accurate duration estimation of such works is important for effective project planning and management in the crowed and congested urban construction sites. Yang [[2]] developed a CBR system for duration estimation of curtain wall construction in his Ph.D. research. Totally 27 historical datasets were collected from major consultant firms of Taiwan. Among which 24 are used for training and 3 are used for testing. The input attributes identified by Yang are: (1) excavation depth; (2) quantity of walls; (3) construction method; and (4) soil type.

The data are shown in Table A.3. Two qualitative attributes are transformed into numeric values: (1) construction methods—1 means ML method, 2 represents MHL; (2) the soil type—Clayey as 1, Sandy-clayey as 2, Sandy as 3, Sandy-gravel as 4, Gravel as 5, and Clayey-gravel as 6. Similar to Example I, the order of data in Table A.2 has been randomized from their original sequence. The last 3 (shaded) datasets of the 27 data in Table A.3 are selected as testing sets. The rest 24 datasets are used for system training. The data incompleteness is simulated by random selections as described in Example I. There are five cases with various degrees of data incompleteness simulated as shown in Table 13.

Table 13. Data incompleteness cases of Ex. III

Measures	Case				
	(1)	(2)	(3)	(4)	(5)
PIA (%)	75	75	100	100	100
PID (%)	17	42	50	58	100
POI (%)	5	10	15	20	25

(2) Testing results

For the first scenario, the DM is performed on 24 complete datasets. The testing results of the two testing data are shown in Table 14. It is found that the average system accuracy is 94.62% for training sets, and 95.94% for the three testing sets.

Table 14. Testing result of complete data—Ex. III

Data		Accuracy (%)
Training sets		94.62%
Testing	Dataset A	95.11%
	Dataset B	94.45%
	Dataset C	98.25%
	Average	95.94%

Similar to Example I, the incomplete data are tested with FALCON by discarding the datasets with missing attribute values for Scenario (2). The numbers of data in training and testing sets for each case are shown in Table 15.

Table 15. No. of training sets for the cases of Ex. III

	Case				
	(1)	(2)	(3)	(4)	(5)
PID (%)	18	32	50	50	100
No. of training sets	20	14	12	10	0
No. of testing sets	3	3	3	3	-

On the other hand, Scenario (3) is tested with different degrees of incompleteness as shown in Table 13 by direct learning on incomplete data. The average accuracy of the three testing sets for the fives cases in Scenario (2) and (3) are shown in Table 16.

It is found from Table 16 that the proposed VaFALCON, Scenario (3), improves the system accuracy significantly compared Scenario (2). While comparing with complete data, the proposed VaFALCON can recover the system accuracy from 87%, for Case (5), up to 93%, for Case (1).

Table 16. Testing results of Ex. II

		System accuracy (%)				
Case		(1)	(2)	(3)	(4)	(5)
Scena-rio	(1)	95.9	95.9	95.9	95.9	95.9
	(2)	85.8	76.6	71.7	66.2	-
	(3)	89.5	85.1	84.3	82.6	83.0
Difference *		3.7	8.5	12.6	16.4	83.0

Recovery**	93%	89%	88%	86%	87%

*The difference of accuracy (%) between Scenario (2) and (3)
**Recovery of accuracy: Acc. of Scenario (2)/Acc. of Scenario (3)

7.5 Summary of system testing

The system testing results obtained from the three testing examples show the excellent capability of the proposed VaFALCON in mining incomplete construction databases. The improvement is outstanding and consistent in all three examples. Following analyzes the performance of VaFALCON from viewpoints of two indexes: (1) accuracy improvement; and (2) accuracy recovery). The analysis is performed with respect to two important incompleteness measures, *PID* and *POI*, defined in Section 3. Another incompleteness index, *PIA*, is not discussed considered here since the *PIA's* of all cases in the three examples are close to 100% without significant variances.

(1) Accuracy improvement of VaFALCON

The improvement of accuracy by VaFALCON is defined as the difference of accuracy between Scenario (2) and Scenario (3). This index shows the benefit of information recovery by VaFALCON from the incomplete data. Figure 9 shows the accuracy improvement of VaFALCON with respect to various percentages of *PID*. While, Figure 10 shows the accuracy improvement of VaFALCON with respect to various percentages of *POI*. It is found that accuracy improvement is increasing as *PID* increases. The DM performance of Scenario (2), traditional data cleaning approaches, degrades dramatically as *POI* exceeds 20% and *PID* approaches 100%. However, the proposed VaFALCON performs consistently well under these conditions.

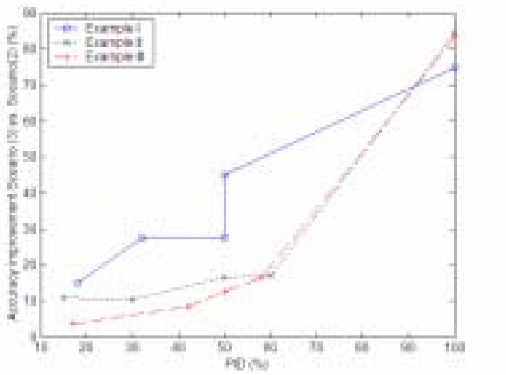


Figure 9. Accuracy improvement vs. *PID*

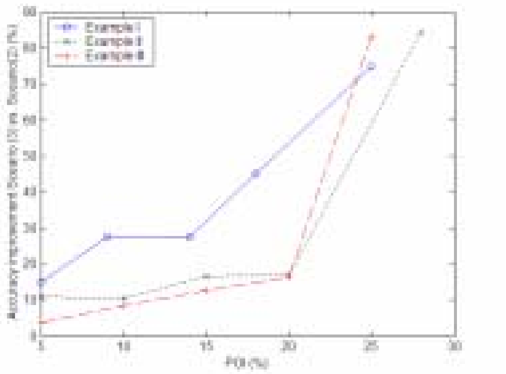


Figure 10. Accuracy improvement vs. *POI*

(1) Accuracy recovery of VaFALCON

Another index, accuracy recovery, indicates the power of VaFALCON to recover the information of the original databases. Figure 11 shows the power of accuracy recovery by VaFALCON vs. various percentages of *PID*. Figure 12 shows the accuracy recovery of VaFALCON vs. various percentages of *POI*. In both figures, it is found that the power of accuracy recovery by VaFALCON decays as the increases of *PID* and *POI*. This is inevitable as the information is leaking with the missing attributes. However, it is noted that, even under the severe situation of Case (5), the power of accuracy recovery of VaFALCON is still greater than 80%. That is, at least 80% of the information of original data is recovered by VaFALCON even when all datasets are incomplete.

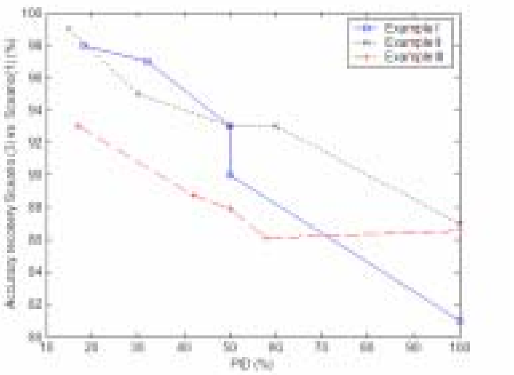


Figure 11. Accuracy recovery vs. *PID*

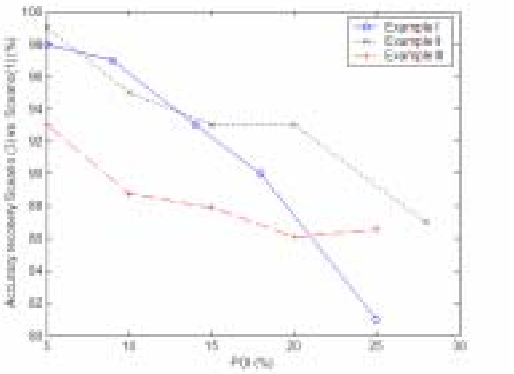


Figure 12. Accuracy recovery vs. *POI*

[1] From analyses of accuracy improvement and accuracy recovery for testing results, it is concluded that the proposed VaFALCON is verified as capable of mining incomplete construction databases. Moreover, the capability of handling incomplete data provides a very powerful feature for DM techniques during usage phase—the application of mined knowledge with incomplete information. VaFALCON is able to provide relatively good solution even when some attribute information of new problem is not complete. This is very common for real time construction problem solving. It is usually impractical to wait until all required attribute values are collected.

六、結論與建議

This research presents the first-of-a-kind variable-attribute numerical data mining technique named VaFALCON. The proposed VaFALCON adopts the structure of a neuro-fuzzy system, so it provides not only functions of numerical mapping but also the explanations of reasoning process and problem trace-back. Moreover, the proposed VaFALCON accepts incomplete construction data with any percentages of incompleteness. It can make best use of the information contained in the incomplete data. From testing results, it is found that the proposed VaFALCON is able to improve the system accuracy up to 84.5% and recover accuracy at least 81% even under severe data incompleteness case, where all datasets of the database are incomplete.

Another very desirable feature of the proposed VaFALCON is its capability to take incomplete information and provide good solutions even when the attribute values of problem domain are not completely collected. Such functions can help decision makers in real-time problem solving.

The proposed VaFALCON is able to handle incomplete data with missing attribute values, however the missing data problem discussed previously is still unsolved. Ambitious researchers are encouraged to pursue in that field.

七、參考文獻

1. Wu, C. F., Yu, W. D., and Yang, J. B. “A Study on the Estimation of Realistic Construction Duration and the Schedule Compression Incentives,” *Report to the Public Construction Commission*, Public Construction Commission, Executive Yuan, Taiwan Government, 2002. (in Chinese)
2. Lin, C. T., and Lee, C. S. G., “Neural-network-based fuzzy logic control and decision system,” *IEEE Transactions on Computers*, Vol. 40, No. 12, pp. 1320-1336, 1991.
3. Ardery, E. R., “Constructability and constructability programs: White paper,” *J. of Constr. Engrg. and Mgmt.*, ASCE, 117(1), pp. 67-89, 1991
4. Yu, W. D., and Yang, J. B., “Data Mining for the Cost Estimating of Highway Bridges Construction with a Neuro-Fuzzy System”, *Proceedings of 2001 Ninth National Conference on Fuzzy Theory and Its Applications*, Nov.

23~24, National Central University, Chung-li, Taiwan, pp. 437~442, 2001.

5. Fayyad, U., and Uthurusamy, R., “Data mining and knowledge discovery in databases,” *Commun. ACM*, vol. 39, pp. 24–27, 1996.
6. Mitra, S., Pal, S. K., and Mitra, P., “Data mining in soft computing framework: A survey,” IEEE Trans. Neural Networks, vol. 13, No. 1, pp. 3–14, 2002.
7. Yu, W. D., and Skibniewski, M. J., “A neuro-fuzzy computational approach to constructability knowledge acquisition for construction technology evaluation,” *Journal of Automation in Construction*, Vol. 8, No. 5, pp. 539-552, 1999.
8. Yu, W. D., and Skibniewski, M. J., “Quantitative constructability analysis with a neuro-fuzzy knowledge-based multi-criterion decision support system,” *Automation in Construction*, Vol. 8, No. 5, pp. 553-565, 1999.
9. Han, J., and Kamber, M., *Data Mining—Concepts and Techniques*, Morgan Kaufmann Publishers, San Diego, U.S.A., pp. 109, 2001.
10. Mamdani, E. H., “Applications of fuzzy set theory to control systems,.” in *Fuzzy Automata and Decision Processes*, Amsterdam, New York, pp. 77-88, 1977.
11. Kohonen, T., *Self-organization and Associative Memory*, Springer-Verlag, Berlin, German, p. 132, 1988.
12. Yu, J. S., “Developing building cost estimating system using case-based reasoning approach,” *Master Thesis*, Department of Civil Engineering, National Central University, Chungli, Taiwan, R.O.C., 2001. (in Chinese)
13. Hsieh, W. S., “Evolutionary conceptual construction cost estimation,” *Master Thesis*, Department of Construction Engineering, National Taiwan University of Science and Technology, Taipei, Taiwan, R.O.C., 2002.

[2] Yang, J. B., “An integrated knowledge acquisition and problem solving model for experience-oriented problems in construction management,” *Dissertation in Partial Fulfillment of Requirements for Degree of Ph.D.*, National Central University, Chungli, Taiwan, R.O.C., 1997.