# SEQUENCE-BASED PREDICTION IN THE CONCEPTUAL DESIGN OF BRIDGES[1]

**Weiyuan Wang and John S Gero**
Key Centre of Design Computing
Department of Architectural and Design Science
University of Sydney
Syndey, NSW, Australia 2006
weiyuan@acslink.net.au
john@arch.su.edu.au

**Summary** This paper explores the application of a machine learning technique in knowledge support systems in civil engineering design. It presents a sequence-based prediction method for engineering design and demonstrates its utility in the conceptual design of bridges. The basic idea of sequence-based prediction is that the most recent numbers of similar design cases are used in predicting the characteristics of the next design and more recent cases are given stronger influence on decision making in the new design situation than older ones. This paper develops a model of sequence-based prediction and carries out a number of experiments using it. It is then applide to a set of standard data and the results of using a sequence-based prediction method are compared with other methods. The empirical results show the potential applications of the method in engineering design.

## 1. INTRODUCTION

Most of the current applications of machine learning techniques in knowledge-based systems in design have focused on the acquisition and maintenance of design knowledge in forms of empirical associations or mappings between different types of design properties (Reich and Fenves, 1991; Wang and Gero, 1993). Although these types of knowledge play important roles in various knowledge support systems in engineering design, it seems impossible to capture design knowledge completely by attribute-value pair or other fixed representation schemes (Coyne and Snodgrass, 1993) due to the dynamic nature of design. New technologies and materials are frequently introduced, existing codes or standards are frequently modified, domain theories are extended or replaced, and social preferences are constantly shifting. A new design can be heavily influenced by the current design practise as well as the

[1]To appear in *ASCE Journal of Computing in Civil Engineering*

designer's previous experiences. It is the authors' conjecture that the chronological sequence of design cases in a domain might hide important information which could reflect the progressive changing of materials, technology, design methods, standards, social preferences, etc. The knowledge hidden in the chronological sequence of design cases could play the same role as empirical associations or mappings between different types of design properties do in the conceptual formulation of new design solutions.

This paper explores the acquisition and application of such knowledge, presents a sequence-based prediction method, and demonstrates its prediction capabilities in the conceptual design of bridges. The basic idea of this sequence-based prediction method can be outlined as follows: the most recent number of similar design cases are used to predict the next design. The more recent design cases are given a stronger influence on decision making in the new design situation. The underlying assumption is that the more recent design experiences are likely to be more significant than older ones and are more likely to be recalled by a designer to assist in a new design. The method has been tested with the Pittsburgh bridges design database obtained from the UCI Repository of Machine Learning Databases and competitive performance results have been achieved.

In the remainder of this paper, Section 2 elaborates the sequence-based prediction model; Section 3 introduces a way to optimize the model by discovering the most suitable parameters of the model; Section 4 demonstrates the application of the method in the conceptual design of bridges; and Section 5 discusses the methods, advances and limitations, related work, and the avenues of future research.

## 2. SEQUENCE-BASED PREDICTION

The sequence-based prediction method introduced here is fundamentally different from the traditional sequence-based prediction or part-to-whole learning (Michalski, 1987) as well as time series prediction (Weigend and Gershenfeld, 1994). Its theoretical foundation is built on the discoveries from psychological studies of human memory (Ebbinghaus, 1885). According to Ebbinghaus's discovery, the retention strength of knowledge stored in memory decreases over time, but the forgetting rate slows down according to a power function. The retention strengths of experiences in memory are reduced by a decay rate or by the interference of storing other experiences (Anderson, 1985). The strength of an abstract concept is reinforced by rehearsal or repeating the related experience (Loftus and Loftus, 1976).

To introduce the method, the concepts of window size and decay rate are explained first with Figure 1. Design cases are identified by their position code in a sequence. The oldest one is 1, the following one is 2, and so on. T is the current position in a sequence when the next case is processed; t is the time passed since a stimulus or case is received (also called retention interval by psychologists), and here it is indicated by the number of cases processed since then. ‘Next case’ is the new design case to be predicted, ‘older cases’ are the previous cases out of the window, the degree of greyness of each cell indicates the remaining strength, called retention strength, of an old stimulus or case at time T. After the next case is processed, the window slides to the right by one case along the chronological sequence of design cases.

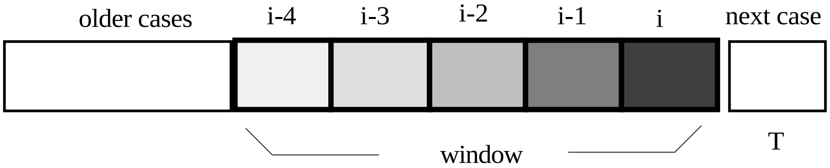


Figure 1. A window slides along a chronological sequence of design cases.

Some other preliminary concepts of the model as well as some useful symbols are introduced as follows:

| | |
|---|---|
| $D$ | decay rate, also called forgetting rate, which means that after a case is incorporated into memory, the retention strength of any item in memory will be reduced by a factor of D |
| $L_{v(i)}(T - t)$ | a logical function: if the value of related attribute of the case at the position T-t matches that of the case at the position T (that is, v(i)), the result is 1; otherwise, 0 |
| $R$ | retention rate, where R = 1 - D, which means that after a case is incorporated into memory, the retention strength of any item in memory will become the product of its old value times R |
| $s_{V(i)}(t)$ | the retention strength of a stimulus, value v(i), after t more cases are incorporated into memory |
| $\mathbf{S}_{V(i)}$ | the accumulated retention strength of the stimulus, value v(i), which is accumulated from all the occurrences of the stimulus so far |
| $\mathbf{S}^{m}_{v(j)}$ | the strongest accumulated retention strength of alternative values |
| $t$ | the time passed since a stimulus or case is received; it is indicated by the number of cases processed since then |
| $T$ | the current position in a sequence when the next case is processed, |

| | |
|---|---|
| | indicated by the position code of the next case to be processed |
| v(i) | the *i*th value of an attribute |
| W | the size of window which indicates the maximum number of items which can be in a window |

A computational model of the sequence-based prediction in design is formally defined as follows:

$$\mathbf{S}_{V(i)} = \sum_{t=1}^{W} s_{v(i)}(t-1) * L_{v(i)}(T-t) \tag{1}$$

where

$$L_{v(i)}(T-t) = \begin{cases} 1 \mid \text{if the attribute value of the case at position T- t is v(i)} \\ 0 \mid \text{otherwise} \end{cases} \tag{2}$$

$$s_{V(i)}(t) = s_{V(i)}(t-1) * R \tag{3}$$

$$s_{V(i)}(t) = s_{V(i)}(0) * R^{t} \tag{4}$$

Let

$$s_{V(i)}(0) = 1 \tag{5}$$

which means that every time when a new case arrives, the strength of the corresponding attribute value will be increased to 1, and which also means that the initial strength of each stimulus is assumed as 1;

then

$$s_{V(i)}(t) = R^{t} \tag{6}$$

Equation (1) is simplified to

$$\mathbf{S}_{V(i)} = \sum_{t=1}^{W} R^{t-1} * L_{V(i)}(T-t) \tag{7}$$

Let $\mathbf{S}^{m}_{v(j)}$ represent the strongest accumulated retention strength of n values of an attribute, which means that the *j*th value has the strongest retention strength (accumulated from all its occurrences up to that time) among all the values of the attribute,

$$\mathbf{S}^{m}_{v(j)} = \mathbf{Max}\ [\ \mathbf{S}_{V(1)},\ \mathbf{S}_{V(2)}, \ldots, \mathbf{S}_{V(j)}, \ldots,\ \mathbf{S}_{V(n)}\ ] \tag{8}$$

then, the predicted value (or default) for the attribute, Attr, is the one with the highest frequency,

$$\text{Predicted-value (Attr)} = v(j) \tag{9}$$

The following examples show how this model works. Suppose the window size, W, equals 5, and there are two types of values A and B, see Figure 2.

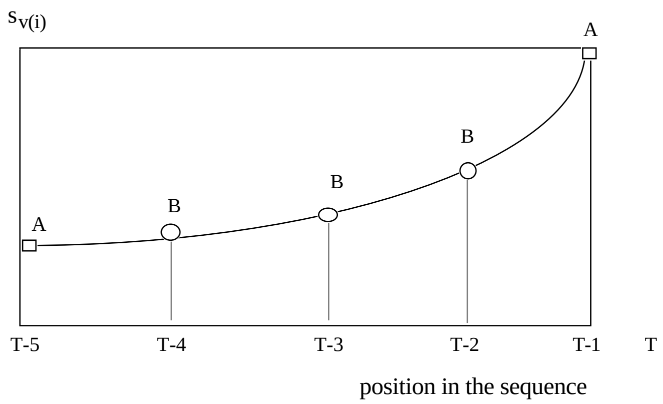


Figure 2. A snapshot of the window with size 5 when processing the case at position T.

The horizontal axis indicates the positions of the five cases in the window and the next case to be predicted. The vertical axis indicates the retention strengths of these five cases in the window.

According to Equation (1), the accumulated retention strength of each value is

$$\mathbf{S}_{V(i)} = \sum_{t=1}^{W} s_{v(i)}(t-1) * L_{v(i)}(T-t)$$

therefore, the accumulated retention strength of value A,

$$\begin{aligned}\mathbf{S}_{A} &= s_A(1\text{-}1)*1 + s_A(2\text{-}1)*0 + s_A(3\text{-}1)*0 + s_A(4\text{-}1)*0 + s_A(5\text{-}1)*1 \\ &= s_A(0) + s_A(4) \\ &= R^0 + R^4\end{aligned}$$

and the accumulated retention strength of value B,

$$\begin{aligned}\mathbf{S}_B &= s_B(1\text{-}1)*0 + s_B(2\text{-}1)*1 + s_B(3\text{-}1)*1 + s_B(4\text{-}1)*1 + s_B(5\text{-}1)*0 \\ &= s_B(1) + s_B(2) + s_B(3) \\ &= R^1 + R^2 + R^3\end{aligned}$$

When R = 1,

$$\mathbf{S}_A = 2$$
$$\mathbf{S}_B = 3$$
$$\mathbf{S}_B > \mathbf{S}_A$$

therefore, the predicted value for next case is B.

When R = 0.5,

$$\begin{aligned}\mathbf{S}_A &= 0.5^0 + 0.5^4 \\ &= 1.0625 \\ \mathbf{S}_B &= 0.5^1 + 0.5^2 + 0.5^3 \\ &= 0.875 \\ \mathbf{S}_A &> \mathbf{S}_B\end{aligned}$$

therefore, the predicted value for next case is A. This example demonstrates the effect of changing the retention rate R.

Let us examine another example. The following subsequence is extracted from the Pittsburgh bridges design dataset.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| completed-date: | 1914 | 1915 | 1915 | 1918 | 1920 | 1921 | 1923 | 1924 |
| type-of-bridge: | simple-t | simple-t | cont-t | simple-t | suspen | arch | arch | suspen |

To predict the type of the next bridge design, the selection of a different window size will give out different results as follows (suppose the retention rate = 1):

| | | | | | |
|---|---|---|---|---|---|
| window size: | 1 | 2 | 3 | 4 | 8 |
| predicted type: | suspen | suspen | arch | suspen | simple-t |

These examples have shown that the predicted value, as well as the predictive accuracy, is dependent on the size of window W and the value of retention rate R.

The implementation of this model of the sequence-based prediction method is simple and the knowledge representation schema is shown in Figure 3. With this knowledge representation, the model can be easily implemented for incremental computation.

```
Schema for sequence prediction

Attr (1)
            value(1)        S(1)
            value(2)        S(2)
             ...

Attr (2)
            value(1)        S(1)
            value(2)        S(2)
            ...
 ...

Attr (n)
            value(1)        S(1)
            value(2)        S(2)
            ...
```

Figure 3 The representation for sequence-based prediction

When predicting the values of attributes of the next case, use Equations (8) and (9) to select the value with the strongest accumulated retention strength $\mathbf{S}^{m}_{v(j)}$ for each attribute. When the next case is incorporated into the schema, update all the accumulated retention strengths by multiplying each of them by the retention rate R. Then, execute the following actions: first, further update the accumulated retention strength of each value that matches the value of the corresponding attribute in the next case (whether or not it is correctly predicted) by the following formula:

$$\mathbf{S'}_{V(i)} = \mathbf{S}_{V(i)} + 1 \tag{10}$$

where $\mathbf{S}_{V(i)}$ is the old accumulated retention strength for v(i) and $\mathbf{S'}_{V(i)}$ is the new one; second, if a case C is pushed out of the window, update the accumulated retention strength of each value that matches a value of the corresponding attribute in case C by the following equation,

$$\mathbf{S'}_{V(i)} = \mathbf{S}_{V(i)} - R^{w-1} \tag{11}$$

where the W is the window size and R is the retention rate. In the implementation, the result of $R^{w-1}$ can be calculated and saved in a table for later look-up to speed up the system's performance.

The process of prediction and knowledge maintenance is summarised below:

1. use Equations (8) and (9) to determine the value of each predicted attribute for the next case;
2. incorporate the case into memory by the following steps,
   - 2.1 update all the accumulated retention strengths by multiplying each of them by the retention rate R;
   - 2.2 by Equation (10), update the accumulated retention strength of each value that matches the value of the corresponding attribute in the next case;
   - 2.3 by Equation (11), update the accumulated retention strength of each value that matches a value in the case that is pushed out of the window.

The model is also cognitively-based in the following sense: in the process of prediction and maintenance, step 1 is an implementation of a competitive recalling or voting mechanism (in accord with inference theory); step 2.1 models the systematic decay mechanism of memory; step 2.2 reflects the effect of rehearsal of knowledge; and step 2.3 simulates the effect of forgotten memories.

## 3. DISCOVERY OF AN OPTIMAL MODEL

An optimal model of sequence-based prediction in a design domain is defined as that which can achieve the highest predictive accuracy with a given sequence of design cases. The 'optimal' is only a hypothesis in a sense that it is true at least up to now. When new cases are included, the performance of the model might *slowly* change. The change of performance could not be fast because the model itself will incrementally learn from new cases. The window slides along the chronological sequence and the strengths are updated and only two parameters W and R in the model are fixed.

The window size W, the retention rate R, and their possible values together define a state space and learning is considered as search through the state space to find an optimal model that produces the best total predictive accuracy in the given domain.

The value range of window size is between 1 and N, where N is the number of design cases available. The value range of retention rate is between 0 and 1. Before the search can start, the value of retention rate must be discretised to some selected interval. For the sake of learning efficiency, the interval is dynamically further divided during the learning process. For example, if the best values up to now are $R_i$ and $W_i$, the range between $R_{i-1}$ to $R_{i+1}$ will be further divided into more smaller intervals, and search will continue in the area defined by the ranges between $R_{i-1}$ and $R_{i+1}$ and between $W_{i-x}$ and $W_{i+x}$ (where x is a selected constant).

An optimal model could be acquired by systematically searching the state space for the optimal values of retention rate and window size to achieve the best prediction. However, even with the step-wise discretization strategy, the systematic search of the whole space could be computationally expensive. This computational complexity can be greatly reduced by qualitatively analyzing the behaviour of the model when changing the retention rate and the window size and by avoiding searching those areas where an optimal model is obviously impossible.

A qualitative analysis of the behaviour of the exponential function in Equation (6) can be used to greatly reduce the search space, Figure 4.

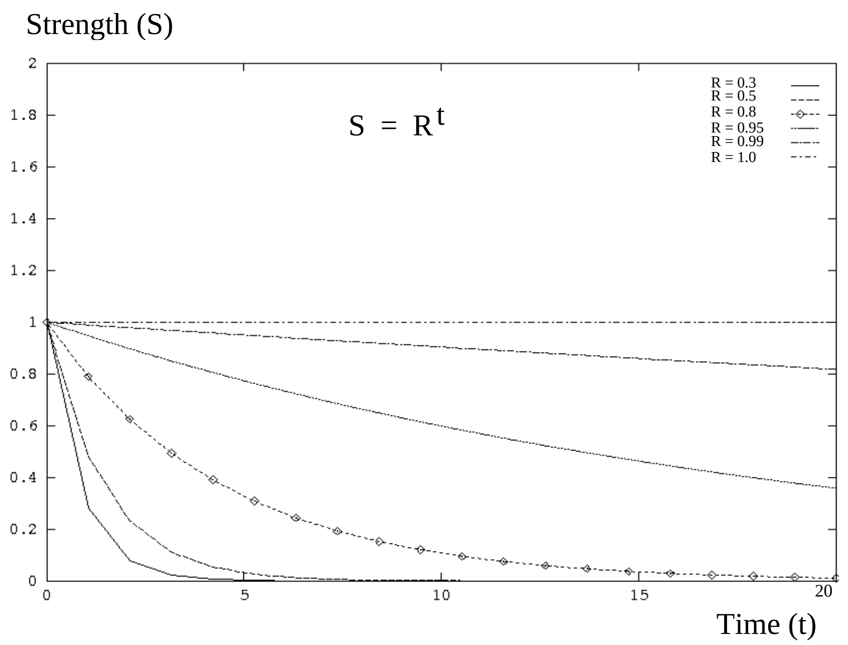


Figure 4. The behaviour of exponential function, $S(t) = R^t$.

Whatever the value of W is, when $R < 0.5$, $R^1 \geq \sum_{t=2}^{W} R^t$. This means that it is unnecessary to search the area where $R < 0.5$. When R increases but its value is smaller than 0.8, only about the 20 first values are significant, as the figure shows. When $t > 20$, $R^t$ becomes very small and can be ignored. This means that for certain values of R, the search is meaningful only in certain range along the axis of window size. It is unnecessary to search the whole range of window sizes from 1 to N

Figure 5 indicates that the prediction accuracy is a function of retention rate. When the retention rate $R = 1$, the model is a simple window-based prediction . When the retention rate $R \leq 0.5$, the prediction is based on the single most recent design case.

Between R > 0.5 and R < 1 there exists some optimal point or points where the prediction accuracy will reach its maximum value.

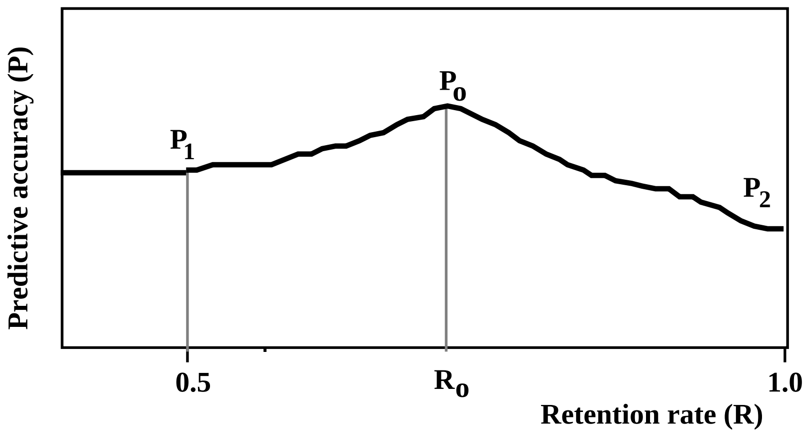


Figure 5. The qualitative analysis of the behaviour of the model when the retention rate R is changed. $P_1$ represents the prediction based on the most recent case; $P_2$ represents the window-based prediction; and $P_o$ represents an optimal prediction, $R_o$ is an optimal retention rate.

Figure 6 shows that the prediction accuracy is also a function of the window size. When window size W = 1, the prediction is actually based on only the single most recent design case. Therefore, there will be no change on prediction accuracy even if the retention rate changes. For a given retention rate, when window size W is equal to or larger than a certain value, $W_m$, there is little influence on the prediction accuracy when the window size further increases, because the strengths of the added cases are too small to be significant. Between W > 1 and W < $W_m$ there exists some point or points where the prediction accuracy reaches its maximum value.

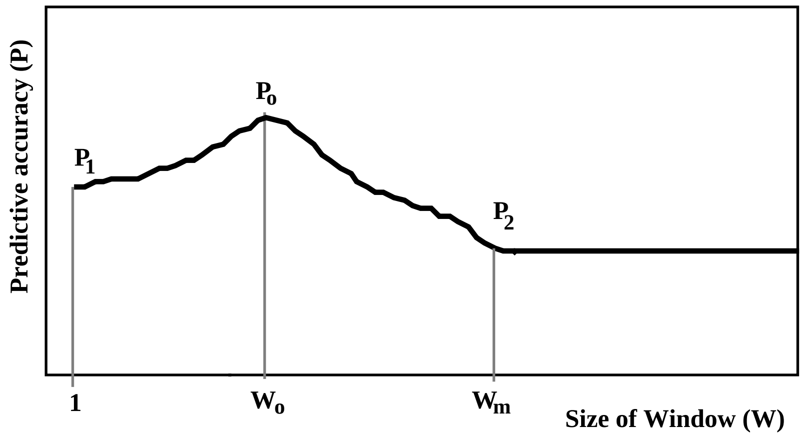


Figure 6 The qualitative analysis of the behaviour of the model when the window size W is changed. $P_1$ represents the prediction based on the most recent case; $P_2$ represents the prediction with a decay rate; and $P_o$ represents an optimal prediction; there is no meaning when W < 1.

## 4. EXPERIMENTAL STUDIES AND RESULTS

The dataset of Pittsburgh bridge design, as mentioned earlier, has been used to test the model. The five design solution attributes are considered in the experiment and they are T-OR-D, MATERIAL, REL-L, SPAN and TYPE. The T-OR-D attribute specifies whether the bridge is a 'through' or a 'deck bridge'; MATERIAL can be wood, iron, or steel; REL-L is the relative length of the main span to the total length; the TYPE describes the type of bridge, which can be one of the six values: 'simple-t' (simple truss), 'conti-t' (continuous truss), 'cantilev' (cantilever), 'suspen' (suspension bridge), arch, and wood. The details of the domain can be found in Reich and Fenves (1991).

Using the systematic and step-wise search strategies, the landscape of the predictive performance of the system for the Pittsburgh bridge design domain has been discovered, and is presented in Figure 7. The highest value of prediction accuracy is 0.705, at the location W = 12 and R = 0.805. In other words, an optimal model (W=12 and R=0.805) of sequence-based prediction in this design has been discovered by the system.

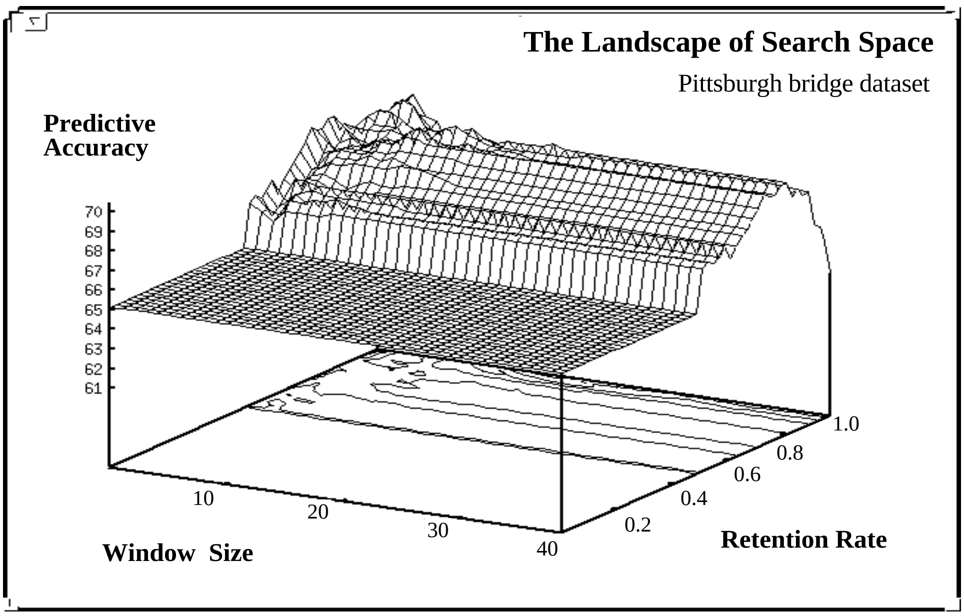


Figure 7. The landscape of the prefidticve performancee discovered by the system for the domain of Pittsburgh bridge designs. The optimal model determined by the system is W = 12 and R = 0.805.

The experiment has indicated that the prediction accuracy is a function of the window size as shown in Figure 8. In the figure, the horizontal line shows the result of prediction based on the most recent single design case, the dark line indicates the result of prediction when R = 1, and the dotted line presents the result of prediction when R = 0.8. It has been discovered that the prediction performance of models where R = 0.8 generally is better than that of a model where R = 1, except when the window size is smaller than 5.

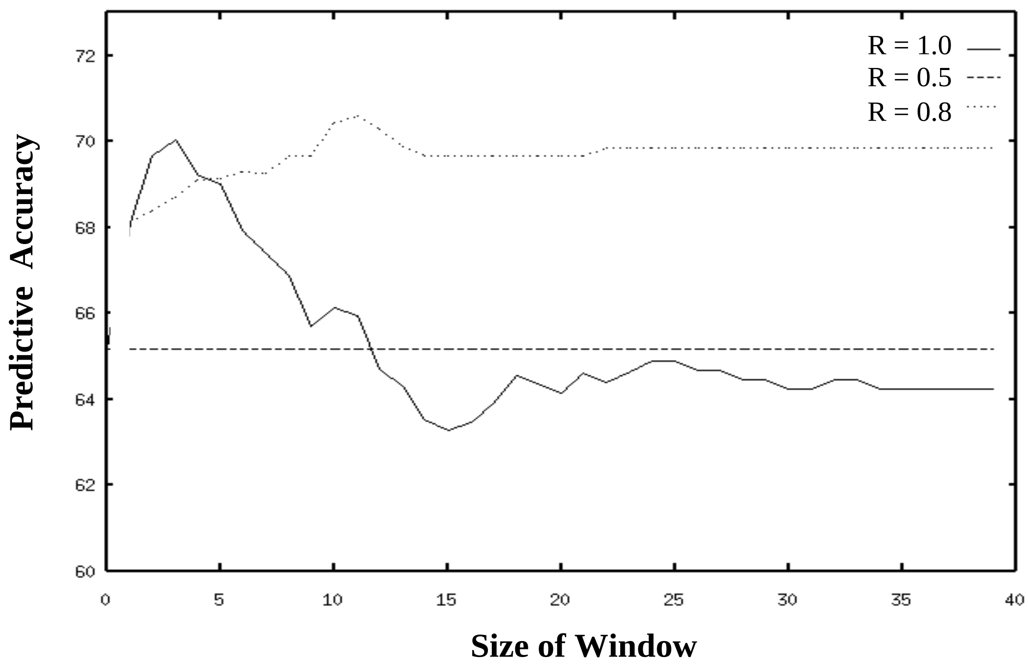


Figure 8. The behaviour of the cognitive model of prediction for the Pittsburgh bridge dataset: the optimal window size is 12.

It has also been shown empirically that the prediction accuracy is a function of the retention rate as shown in Figure 9. The horizontal line between R = 0 and R = 0.5 indicates that the prediction accuracy does not change even if the window size is changed. It has been discovered that the predictive accuracy of models where W = 12 generally is better than that of others except when the retention rate R is close to 1. When W = 3, the best result is close to the point where R = 1, which means that when the size of the window is very small, a better result may not be obtained by reducing the value of R.

Both Figures 8 and 9 indicate that for incremental frequency-based prediction (R = 1 and W ≥ 40), the predictive accuracy is about 64%; using the most case (R ≤

0.5), the predictive accuracy is about 65%. The best predictive accuracy of this prediction model is 70.5%, which is higher than that of Bridger (64.6%).

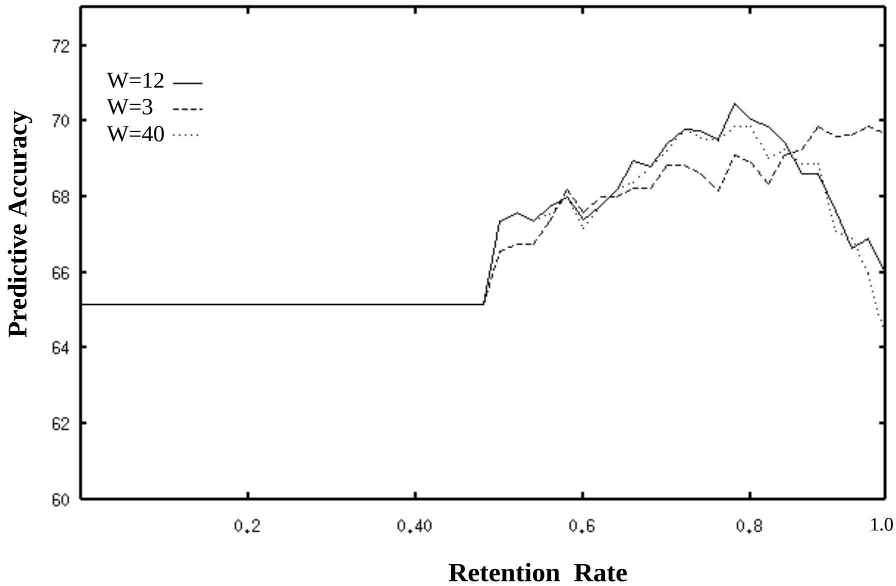


Figure 9. The behaviour of the cognitive model of prediction for the Pittsburgh bridge dataset: the optimal retention rate is 0.805.

A sample of the predictive activities is presented in the Appendix. These predictions show a gradual shift over the period 1818 and 1878 from bridges where the type of material dominates (wood), to where the type of structure dominates (simple trusss bridge type) in the period 1872 to the 1920s. Later, the cantilever style bridge type began to dominate, followed by the arch and finally, this leads to the continuous truss bridge. This clearly demonstrates a benefit of using a sequence-based prediction method as it gives predictions over a time sequence. A traditional machine learning approach would produce only one prediction across all cases.

What is suggested by these experiments is that the sequence-based prediction might achieve a predictive accuracy as good as similarity-based prediction does in the domain of design and its integration with other machine learning methods might produce a higher prediction performance.

Such an integration is demonstrated with the following example, in which concept formation and sequence-based prediction are combined (Wang, 1994). A decision tree was created with the attributes 'lane' and 'erected'; then, a sequence-based prediction method was used at each node. One of the results is shown in Figure 10, where CC presents the number of cases in that branch and PA indicates the predictive accuracy of that node. The total predictive-accuracy obtained in this experiment is 74.2%, which is higher than that produced by a symbolic machine learning method.

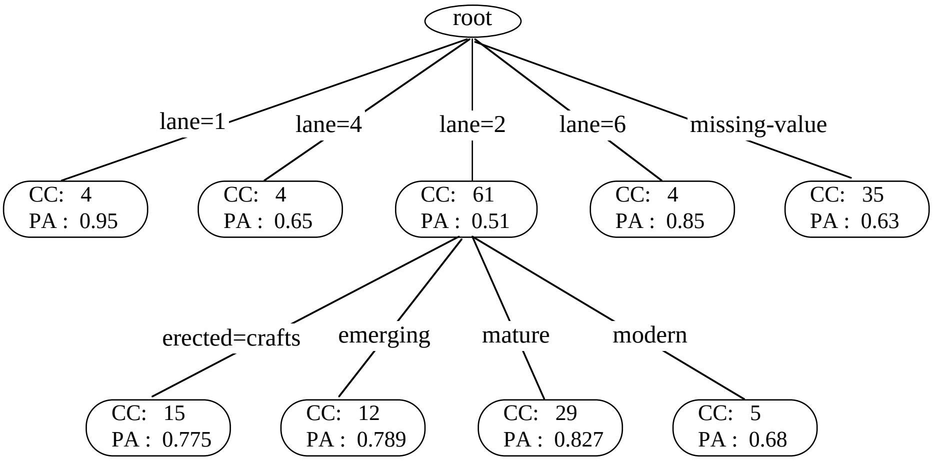


Figure 10. A results produced by an integration of sequence-based prediction with concept formation

## 5. DISCUSSION

Most incremental concept learning systems are order dependent. Such ordering effects have been widely mentioned in the literature (Cornuejols, 1993). The ordering effects mean that given a set of cases, differently ordered sequences of these cases lead to different learning results. The ordering effects in incremental learning have been studied by MacGregor (1988) and others. They assumed that the learning algorithms are satisfactory and the order of examples or observations is adjustable, and attempted to find the optimal order to get the best learning result. Some researchers have tried to make a learning algorithm order-independent at the expense of increased memory and/or computational time such as in ID5.

In instance-to-class learning (either supervised or unsupervised), it is assumed that the order of examples or observations are adjustable but the parameters of a learning model are fixed. This is due to the underlying assumption that there is no knowledge in the sequence of examples. In some incremental learning systems, work has been carried out to avoid the order effects or to find the best order of a given dataset to achieve the best performance (Cornuejols, 1993).

In sequence-based prediction, in contrast, it is assumed that the order of observations are fixed and the parameters of a learning systems are adjustable. Learning includes finding the best parameters for the learning model to achieve the best performance in prediction.

At the beginning of this paper, the power function, Equation (12), was used instead of the exponential function in Equation (6) in the model for sequence-based prediction in design. The experiment was conducted and slightly better results were achieved.

$$s_{V(i)}(t) = t^{-E} \tag{12}$$

Anderson (1990) has used the following, Equation (13), in the development of computational model of the origins of human knowledge:

$$S = \sum_{i=1}^{n} t_i^{-d} \tag{13}$$

where S is the accumulated strength of a particular item of knowledge, the summation is over the n uses of the item, each $t_i$ is the time since the use of the item, the exponent d is a constant.

A difficulty associated with these models in Equations (12) and (13) is that they are not efficient for incremental learning. To simplify the computational complexity at performance time, for each i, a value of $t_i^{-d}$ can be calculated and the results stored in a look-up table in advance. In this way, the speed of the system can be greatly improved.

By analysing and comparing the behaviours of the power function and the exponential function, it is found that their behaviours are similar when $t > 1$ and $R < 0.9$. After a number of experiments was conducted, the conclusion could be drawn that for the purpose of modelling the information in the sequence of design cases, a similar prediction accuracy can be achieved by using either the power function or the exponential function. The advantage of the exponential function is that it can be calculated incrementally, which is most suitable for incremental learning.

The process of determining an optimal model is non-incremental. However, after the model is determined, the maintenance of the model during its application is incremental and automatic. Another advantage of this model is that little memory is required for the representation of acquired knowledge. The main disadvantage of this approach is that the determination of an optimal model is computationally expensive, irrespective of whether an exponential function or a power function is used.

It can be concluded from this study that many designs are hard to predict by logic or causal models. Sequence-based prediction can perform better than similarity-based predictions in some design domains for a given parameter. An integration of sequence-based prediction and concept formation is well suited for the development of machine learning algorithms in the design domain.

**ACKNOWLEDGEMENT**

This work was supported by a Sydney University Postgraduate Research Award and an Australian Research Council grant. The authors wish to acknowledge the advice of the paper referees.

## REFERENCES

Anderson, J.R. (1985). *Cognitive Psychology and Its Implications*, W.H. Freeman and Company, New York.

Anderson, J.R. (1990). A theory of the origins of human knowledge, *in* J. Carbonell (ed.), *Machine Learning*, MIT, Cambridge, MA, pp.313-352.

Cornuejols, A. (1993). Getting order independence in incremental learning, *Training Issues in Incremental Learning, AAAI Spring Symposium* '93, Stanford University.

Coyne, R.D. and Snodgrass, A. (1993). Rescuing CAD from rationalism, *Design Studies* **14** (2): 100-123.

Ebbinghaus, H. (1885). *Memory: A Contribution to Experimental Psychology,* (translated by H.A. Ruger and C.E. Bussenues, 1913), Teachers College, Columbia University, New York.

Loftus, G.R. and Loftus, E.F. (1976). *Human Memory: the Processing of Information,* Lawrence Erlbaum Associates, Hillsdale, New Jersey.

MacGregor, J. (1988). The effects of order on learning classifications by example: heuristics for finding the optimal order, *Artificial Intelligence* **18**: 203-226.

Michalski, R. (1987). Laerning strategies and automated knowledge acquisition – an overview, *in* L. Bolc (ed.), *Computational Models of Learning,* Springer-Verlag, Berlin, pp.1–19.

Reich, Y. and Fenves S. J. (1991).The formation and use of abstract concepts in design, *in* D. H. Fisher, M. J. Pazzani and P. Langley (eds), *Concept Formation: Knowledge and Experience in Unsupervised Learning*, Morgan Kaufmann, San Mateo, CA.

Wang, W. and Gero, J.S. (1993). Acquiring design knowledge from design cases, *Artificial Neural Networks and Expert Systems*, IEEE Press, Los Alamitos. pp.338-240.

Wang, W. (1994). *A Method and Application of Machine Learning in Design*, PhD Thesis, Department of Architectural and Design Science, University of Sydney, Sydney.

Weigend, A. S. and Gershenfeld, N.A. (eds) (1994). *Time Series Prediction: Forecasting the Future and Understanding the Past* , Addision-Wesley Publishing, Reading, MA.

**APPENDIX: Samples of Predictions of Conceptual designs for Pittsburgh Bridges.**

| period | actual-type | predicted-type |
|---|---|---|
| 1818 | wood | ---- |
| 1819 | wood | wood |
| 1829 | wood | wood |
| 1837 | wood | wood |
| 1838 | wood | wood |
| 1840 | wood | wood |
| 1844 | suspen | wood |
| 1846 | suspen | wood |
| 1848 | wood | wood |
| 1851 | wood | wood |
| 1853 | wood | wood |
| 1856 | wood | wood |
| 1856 | wood | wood |
| 1857 | wood | wood |
| 1859 | suspen | wood |
| 1863 | simple-t | wood |
| 1864 | simple-t | wood |
| 1866 | wood | wood |
| 1870 | wood | wood |
| 1874 | simple-t | wood |
| 1876 | suspen | wood |
| 1876 | wood | wood |
| 1878 | simple-t | wood |
| 1882 | simple-t | simple-t |
| 1883 | simple-t | simple-t |
| 1883 | simple-t | simple-t |
| 1884 | simple-t | simple-t |
| 1884 | suspen | simple-t |
| 1884 | arch | simple-t |
| 1887 | simple-t | simple-t |
| 1887 | simple-t | simple-t |
| 1888 | simple-t | simple-t |
| 1889 | simple-t | simple-t |
| 1890 | simple-t | simple-t |
| 1890 | simple-t | simple-t |
| 1891 | simple-t | simple-t |
| 1891 | simple-t | simple-t |
| 1892 | simple-t | simple-t |
| 1892 | wood | simple-t |
| 1893 | simple-t | simple-t |
| 1894 | simple-t | simple-t |
| 1895 | simple-t | simple-t |
| 1896 | suspen | simple-t |
| 1896 | arch | simple-t |
| 1897 | simple-t | simple-t |
| 1987 | simple-t | simple-t |
| 1898 | simple-t | simple-t |
| 1900 | simple-t | simple-t |
| 1900 | simple-t | simple-t |
| 1901 | simple-t | simple-t |
| 1902 | cantilev | simple-t |
| 1903 | simple-t | simple-t |
| 1903 | simple-t | simple-t |

| period | actual-type | predicted-type |
|---|---|---|
| 1903 | simple-t | simple-t |
| 1903 | simple-t | simple-t |
| 1904 | simple-t | simple-t |
| 1904 | simple-t | simple-t |
| 1904 | cantilev | simple-t |
| 1908 | simple-t | simple-t |
| 1909 | simple-t | simple-t |
| 1909 | simple-t | simple-t |
| 1910 | cantilev | simple-t |
| 1911 | cantilev | simple-t |
| 1914 | simple-t | simple-t |
| 1914 | simple-t | simple-t |
| 1915 | simple-t | simple-t |
| 1915 | cont-t | simple-t |
| 1918 | simple-t | simple-t |
| 1920 | suspen | simple-t |
| 1921 | arch | simple-t |
| 1923 | arch | simple-t |
| 1924 | suspen | arch |
| 1926 | suspen | suspen |
| 1926 | cantilev | suspen |
| 1927 | arch | suspen |
| 1927 | cantilev | arch |
| 1927 | cantilev | cantilev |
| 1927 | arch | cantilev |
| 1928 | cantilev | arch |
| 1928 | suspen | cantilev |
| 1928 | simple-t | cantilev |
| 1928 | arch | cantilev |
| 1931 | arch | arch |
| 1931 | suspen | arch |
| 1931 | cont-t | arch |
| 1937 | cantilev | arch |
| 1939 | cont-t | cantilev |
| 1945 | simple-t | cont-t |
| 1945 | cont-t | simple-t |
| 1945 | cantilev | cont-t |
| 1945 | arch | cantilev |
| 1945 | simple-t | arch |
| 1945 | simple-t | simple-t |
| 1950 | cont-t | simple-t |
| 1951 | cont-t | cont-t |
| 1951 | cantilev | cont-t |
| 1951 | cont-t | cont-t |
| 1955 | simple-t | cont-t |
| 1955 | cont-t | cont-t |
| 1959 | arch | cont-t |
| 1959 | cont-t | cont-t |
| 1961 | cont-t | cont-t |
| 1962 | arch | cont-t |
| 1969 | arch | cont-t |
| 1975 | arch | arch |