Temporal Decision Tree and Interpretable Temporal Rules: J48 and Fuzzy Cognitive Maps Approach

Shih Yin Ooi¹,*, Shing Chiang Tan¹, Wooi Ping Cheah¹

¹Faculty of Information Science and Technology, Multimedia University
Jalan Ayer Keroh Lama, 75450 Melaka, Malaysia
{syyooi, sctan, wpcheah}@mmu.edu.my

Abstract. Temporal data classification is relatively young in the area of machine learning and data mining, where the historical data are used to predict the future value. A nonlinear temporal data classification is proposed in this work, namely Temporal-J48. With its tree-based architecture, the implementation is relatively simple. The classification information is readable through the generated temporal rules. The interrelationship among the generated rules also made available to the user through Fuzzy Cognitive Maps (FCM).

Keywords: temporal data, classification, temporal decision tree, temporal sequences, data mining.

1 Introduction

The conventional data classification is not suitable for a situation where the occurring effects (outcomes) are delayed, i.e.: in the application of weather forecasting, the prediction of air temperature can achieved higher accuracy if considering some values from the wind speed which are collected on some hours ago. Temporal data classification is an area to classify the value of a class attribute based on the values of other attributes by taking advantage of the inherent sequence in the records [1]. Thus, the observed data can be collected from different time periods or in a sequence of events. In this work, we propose a new nonlinear temporal classifier with interpretable temporal rules through the decision tree and FCM approach.

The organization of this paper is as follows. In Section 2, some related literature reviews on the temporal data classification are presented. In Section 3, the methodology of this work is justified. In Section 4, some experimental results and benchmark comparisons on the proposed method are discussed. Lastly, Section 5 concluded our work.

2 Related Works

Many related works have been done to solve the temporal data issue in certain specific applications, especially through the extension of ordinary classifiers, including statistical methods [2], neural networks [3,4,5], and similarity-based techniques [6]. Most of these classifiers are presented with high performance accuracy. However, the classification information and reasoning which are important to understand the inherent classification characteristics are not made available to the users.

One of a few studies related to the temporal data classification has been published by [7]. The method mainly integrates the sequence mining and the Naïve Bayes classifier to reduce the feature dimension, and named after Feature-Mine (FM). Tseng and Lee [8] proposed another classification method, classify-by-sequence (CBS) by using sequential patterns in year 2005 and the work was extended in year 2009 [9]. CBS is an integration method of sequential pattern mining and probabilistic induction. The classification information was made readable through the scoring policy [10]. Karimi et al. [11] proposed a tool to generate temporal rules from the sequential data, namely as Time Sleuth (maintained since 2001 until present). However, the focus of the tool is to discover the causality and acausality of the temporal rules instead of classification and unable to process a large dataset. Revesz et al. [12] proposed a new linear temporal classification method based on decision trees and support vector machines. Their method was outperformed the standard classification methods on two applications: (i) weather forecasting, and (ii) influenza. However, the linear temporal classification method may suffer from poor predictive performance when the observed relationships are non-linear.

3 Methodology

3.1 Temporal Tree

In the temporalization procedure, a temporalized record can be formed by merging consecutive records (data flattening) in a dataset by using a time window of \( w \). For instance, considering a dataset with 4 temporally consecutive records as
shown in Table 1, each with 3 attributes. By flatten these records using a window size of 2, the temporalized records can be generated.

<table>
<thead>
<tr>
<th>Original Records (n=1)</th>
<th>Temporalized Records (n=2)</th>
</tr>
</thead>
<tbody>
<tr>
<td>x  y  z</td>
<td>x (t1) y (t1) z (t1) x (t2) y (t2) z (t2)</td>
</tr>
<tr>
<td>0  0  0</td>
<td>0 0 0 0 0 0 0 1 1 1 0 0 1 1 1 0 0 0</td>
</tr>
<tr>
<td>1  0  0</td>
<td>1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>1  1  0</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
<tr>
<td>0  0  0</td>
<td>0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0</td>
</tr>
</tbody>
</table>

By arranging the records in such way, the existing non-temporal classifiers such as C4.5 [13] can be used [11]. In this work, the temporal tree is modified based on the latest research version of C4.5 approach (or known as C4.8), and implemented as J48 in Weka package [14]. To ensure that no node to have an unknown value during classification, we will ignore the missing data during the information gain calculation instead of using the corrected gain ratio criteria as in the original work [15].

For J48, the last variable (z in this case) will always be deemed as decision attribute. One of the rules that can be extracted from the above example (for original records) would be: if \( (x=1) \) AND \( (y=1) \) then \( z=1 \). To adopt this in temporal fashion, we modified the J48 algorithm in such a way that, all attributes must carry along their respective time label based on the selected window size, \( w \). This is to ensure that during the step of choosing the best condition attribute for splitting, the time criterion will also be considered. In this way, the condition attributes will be ranked based on their temporal order as well as their suitability for expanding the tree. For an instance, if a condition attribute with time label \( t1 \) is used at a node, then its children can only use the condition attributes with a time label \( t2 >= t1 \). In this way, the temporal rule extracted will be: if \( \{ (x_{t1} = 1) \) AND \( (y_{t2} = 1) \) AND \( (z_{t1} = 1) \) AND \( (x_{t2} = 0) \) AND \( (y_{t2} = 0) \} \) then \( z_{t2} = 0 \), where \( z_{t2} \) is the decision attribute.

The discovery of temporal relations within the selected \( w \) time steps for a temporal dataset has been made possible and easily interpretable through the generated temporal rules. However, the generated temporal rules are independent to each other, and have no connection with each other.

With our best knowledge, there are no research teams exploited on this matter before. According to [16], Fuzzy Cognitive Maps (FCM) has the ability to reconstruct the premises behind the behavior of given agents, and providing the explanation for their decisions. Thus, we see the potential to adopt this concept in discovering the interrelationship among the temporal rules which formed the temporal tree.

### 3.2 FCM in Discovering Interrelationship among Temporal Tree Rules

In this work, FCM model is used to explain the interrelationship among the temporal rules. The potential of a FCM model is highly determined by the method used. Over the past decades, many researchers were utilizing FCM in various scientific areas. In order to handle the data from diverse databases, many researchers attempted the situation by using different calculation methods [16]. In this work, we adopted the equation proposed by Stylios and Groumpos [17], to calculate the values of the rules (concepts):

\[
V_j^t = f \left( \sum_{i=1}^{n} k_i^j V_i^{t-1} W_i + k^j V_j^{t-1} \right), \quad 0 \leq k_i^j \leq 1, 0 \leq k^j \leq 1
\]

(1)

where, \( V_j^t \) is the value of rule \( R_j \) in the current observation, \( V_i^{t-1} \) is the value for rule \( R_i \) in the previous observation, and \( V_j^{t-1} \) is the value of rule \( R_j \) in the previous observation. In this work, the generated temporal rules from the temporal decision tree are the main factors to describe the behaviors of the application. Therefore, instead of representing a node with an attribute (as in original FCM contribution), we represent a node with a single temporal rule \( R \) (i.e. if the Temporal-J48 generated 100 rules, then we will have 100 nodes). We set the coefficient \( k_i^j \) to 1 to indicate the influence of the interconnected rules is always high. Instead of allowing the field expert to determine the value of coefficient \( k_i^j \) [17], we replace the value of \( k_i^j \) with the gain value of the final node for each observed temporal rule. The gain value for each temporal rule is obtained through the Temporal-J48 training process, and can be used to represent the proportion of their contribution based on the historical data respectively. Hence, the coefficient \( k_i^j \) has a different value from rule to rule. Since all of the temporal sequences have been addressed by the Temporal-J48, the generated temporal rules are ready to feed directly into a FCM model.
FCM is a bivalent state, thus, to denote the degree of each fuzzy value \( y_j^t \) in the temporal tree, it needs to be squashed into a normalized range \([0, 1]\) in order to represent two decisions of \([\text{false}, \text{true}]\) through a threshold function \( f \):

\[
f(x) = \frac{1}{1 + e^{-\lambda x}}, \lambda > 0
\]  

(2)

\( W_{ij} \) is the weight of the interconnection from rule \( R_i \) to rule \( R_j \). In the original contribution by [17], the weight is determined by the field expert, whereas in our work (with the absence of field expert), we set the weight and the corresponding effect (either “positive” or “negative”) of one rule on the others from the perspective of machine learner by using the following schemes:

**Case 1: Positive causal interrelationship**: when there is at least one \((>0)\) similar attribute chosen for a pair of observed temporal rules, and their final decisions are similar

In this proposed methodology, the node is representing a temporal rule instead of a single attribute, thus, it is important to note that the rule itself may carry multiple chosen attributes. Therefore, if the set of compared rules is leading to the same final decision, the sign will be set to positive (+) to indicate these pair might have positive effect on each other. In order to calculate the interconnection weight \( W_{ij} \), every chosen attributes for rule \( R_i \) and rule \( R_j \) will be compared, and the ratio of how often the same chosen attribute appeared in both observed temporal rules, \( R_i \) and \( R_j \) will be calculated. For example:

\[
R_i : x_{11}, y_{12}, z_{13} \rightarrow \text{ClassA}
\]

\[
R_j : x_{12}, y_{12}, z_{14} \rightarrow \text{ClassA}
\]

In these pair, there is one similar attribute has been chosen in both observed temporal rules, which is \( y_{12} \), so the weight \( W_{ij} \) will be \( \frac{2}{6} = 0.3333 \), this value represents the influential value of \( R_i \) to \( R_j \). The weighted arc for these pair would then be +0.3333.

**Case 2: Negative causal interrelationship**: when there is at least one \((>0)\) similar attribute chosen for a pair of observed temporal rules, and their final decisions are contradict

In contrast, if the set of compared rules is leading to different final decisions, the sign will be set to negative (-) to indicate these pair might have negative effect on each other. The same calculation of the interconnection weight \( W_{ij} \) as in case 1 will be used. For example:

\[
R_i : x_{11}, y_{12}, z_{13} \rightarrow \text{ClassA}
\]

\[
R_j : x_{12}, y_{12}, z_{14} \rightarrow \text{ClassB}
\]

In these pair, there is one similar attribute has been chosen in both observed temporal rules, which is \( y_{12} \), so the weight \( W_{ij} \) will be \( \frac{2}{6} = 0.3333 \), this value represents the influential value of \( R_i \) to \( R_j \). The weighted arc for these pair would then be -0.3333.

**Case 3: No interrelationship**: when there is no \((=0)\) similar attribute chosen for a pair of observed temporal rules

Either leading to same decision (“positive” effect) or contradict decision (“negative” effect), if the chosen attributes for both observed temporal rules are completely different and not overlapping, then the results would be +0 or -0, both indicating there are no relationship exist among them. It is not practical to discover the influential value (dependency) among two temporal rules which do not shared any similar attributes (at least from the perspective of machine learner).

Based on the three aforementioned schemes, fuzzy rule value \( y_j^t \) can be calculated for each temporal rule. From these values, the contributions of each temporal rule in the generated temporal tree are recorded. In this methodology, we complement the proposed Temporal-J48 in such a way that, instead of providing a classification result, the temporal rules and the explanation of each temporal rule are made readable. We believed that these data will be beneficial to the field expert for further interpretation.
4 Experimental Evaluation

To evaluate the proposed method, we use the ozone level detection dataset from UCI Machine Learning Repository [18], which recorded ground ozone level data between 1998 and 2004 at the Houston, Galveston, and Brazoria area. There are two detection datasets included in this collection: (i) 1-hour peak set, and (ii) 8-hour peak set. This dataset is multivariate, continuous, sequential and time series. The datasets have 73 attributes and 2536 instances. Some instances consist of missing values. The following procedure is used to predict the class:

i. For the 1-hour peak set (year 1998-2001): the data from year 1998-2000 are used as a training set, and the data in year 2001 are used as a predicting (testing) set;
ii. For the 8-hour peak set (year 1998-2004): the data from year 1998-2003 are used as a training set, and the data in year 2004 are used as predicting (testing) set.
iii. 10-fold cross validation is used for both of these datasets.

4.1 Experimental Results

1-hour peak set (73 attributes, 1356 instances). According to the prediction results as shown in Fig. 1, the robustness of the classifier based on all 72 attributes is considered promising. By using the proposed algorithm of Temporal-J48, the result for window size = 1 (no temporal value) is better than window size = 2, 3, and 4, this is because when the window size is > 1 (when temporal value is considered), the tree tends to become smaller and this leads to the increasing of the tree’s error rate. However, we observed that the error rates are reduced while moving from window size of 4 to 7 (when more windows are considered). This is very interesting because it implies that there are relationships between the current values of the decision attribute and the current and previous values of the condition attributes. The performance accuracy of Temporal-J48 (when \( \text{w}_{\text{best}} = 7 \)) is even higher than an ordinary J48 (when \( \text{w} = 1 \)). This phenomenon substantiated our argument on the delayed temporal effects.

8-hour peak set (73 attributes, 2534 instances). We observed the similar pattern of performance for 8-hour peak set in Fig. 1, but with higher performance rate. This finding shows the linear relationship between the number of instances and the Temporal-J48 performance result. When the size of instances is increased, the result is greatly improved. The performance accuracy of Temporal-J48 (when \( \text{w}_{\text{best}} = 7 \)) is even higher than an ordinary J48 (when \( \text{w} = 1 \)). Once again, this phenomenon substantiated our argument on the delayed temporal effects.

4.2 Performance Comparison

In order to validate the robustness of Temporal-J48 in the data classification, we performed the same testing scheme by using another two benchmark classification methods, which are (i) radial basis function (RBF) network, and (ii) Multi-layer Perceptron. The WEKA package [14] is used to implement these two methods. The summary of results is shown in Table 2 for 1-hour peak set ozone dataset, and Table 3 for 8-hour peak set ozone dataset respectively. The results from the proposed method are very encouraging. As can be seen from the Table 2 and Table 3, Temporal-J48 can perform better than RBF network and Multilayer Perceptron when dealing with temporal dataset.
Table 2. Prediction accuracy of Temporal-J48, RBF network, and Multilayer Perceptron for 1-hour peak set database

<table>
<thead>
<tr>
<th>Measurement</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Predicted</td>
<td>265</td>
<td>98.15</td>
<td>261</td>
<td>96.67</td>
<td>257</td>
<td>95.19</td>
</tr>
<tr>
<td>Incorrectly Predicted</td>
<td>5</td>
<td>1.85</td>
<td>9</td>
<td>3.33</td>
<td>13</td>
<td>4.81</td>
</tr>
</tbody>
</table>

Table 3. Prediction accuracy of Temporal-J48, RBF network, and Multilayer Perceptron for 8-hour peak set database

<table>
<thead>
<tr>
<th>Measurement</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
<th>Instance Size</th>
<th>Percentage (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Correctly Predicted</td>
<td>364</td>
<td>99.45</td>
<td>355</td>
<td>96.99</td>
<td>344</td>
<td>93.99</td>
</tr>
<tr>
<td>Incorrectly Predicted</td>
<td>2</td>
<td>0.55</td>
<td>11</td>
<td>3.01</td>
<td>22</td>
<td>6.01</td>
</tr>
</tbody>
</table>

5 Conclusion

Through the experimental testing on the dataset of ozone level detection, the proposed Temporal-J48 spells four advantages: (1) it is easy to implement due to the simplicity of decision tree methodology, (2) higher classification accuracy in classifying temporal data if compared to ordinary decision tree, (3) all classification information are readable by human, and easily interpretable through the generated temporal rules, and (4) knowledge discovery for interrelationship among the generated temporal rules.

For future work, we would like to investigate on the usefulness of the extracted fuzzy rule value from each temporal rule, i.e.: extension or perfecting the current temporal tree model, temporal rules reduction, and complementary of the information gain calculation during a temporal tree formation.

References


