2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Extracting Temporal Knowledge from Time Series: A Case Study in Ecological Data

Reggio N Hartono, Russel Pears, Nikola Kasabov
Knowledge Engineering and Discovery Research Institute
Auckland University of Technology
Auckland, New Zealand
Email: reggionh@hotmail.com

Susan P Worner
Bio-Protection Research Centre
Lincoln University
Canterbury, New Zealand

***Abstract*—This research presents a generic framework and methods for mining temporal rules from multiple time-series data and its application to ecological data. The aphids dataset that tracks the trajectory of aphid infestations over time has been well researched in a number of studies. Those studies concentrated on predicting the scale of infestation over time. The focus of our research is to identify environmental factors that predict, in a temporal fashion, high incidence of aphid activity. This required the development of a novel framework for knowledge extraction from multiple time-series data and a method for discretization of numeric data as well-known methods such as SAX did not perform adequately due to the non-Gaussian nature of the data involved. Our experimentation yielded new insights into the environmental factors that may influence pest outbreak which are captured in the form of simple actionable rules that would be of interest to the farming community.**

I. INTRODUCTION

Time-series data is ubiquitous in various scientific fields, and analyzing time-series data is an active area of research [1]. Although intensely researched from the numeric prediction perspective relatively less research has focused on extraction of temporal knowledge in the form of sequential rule discovery. In an ecological context, knowledge of the environmental factors associated with a pest invasion assumes equal if not more importance than mere prediction as it enables end users to make timely decisions on when to put in place suitable pest control measures. The aphids dataset [2] provides a rich source of information on numbers of aphids collected over a period of 20 years. The trap is used as an indicator of aphid abundance in nearby wheat crops. The grain crops are at risk from serious viruses that can cause devastating economic damage that is transferred to the plants when aphids feed on them. Environmental factors such as temperature, wind speed, humidity and other factors that are potentially conducive to the growth of the aphid population are also available. Whilst a number of studies have been conducted on this particular dataset [3] and pest invasion in general [4][5], most of them have been from the perspective of using these variables to predict annual aphid abundance and none have addressed the issue of extracting knowledge about pest outbreaks in the form of temporal rules. This motivates the current research.

In general, knowledge extraction from temporal data in the form of sequential rules from time series requires data pre-processing to transform numeric data into nominal form suitable for application of rule extraction methods. While a number of discretization methods are available for time series data, none of them were found suitable for this study due to the highly skewed nature of species count time-series and this required us to develop a new discretization method that is generic in the sense that it does not assume that the underlying data follows a particular distribution such as the Gaussian as the well-known SAX method [6] requires.

In this study, we also describe a framework for sequential rule extraction from multiple time series data that incorporates discretization; pattern specification for rule generation, sequential rule mining, and finally rule evaluation methods.

Our empirical study on the aphids dataset revealed that high confidence rules that predict an impending aphid outbreak can be identified that gives growers an adequate window of time to take preventive action. The knowledge encapsulated by such rules could not be deduced by simple visualization methods due to the complex inter-relationships between the variables, thus reinforcing the need for application of rule mining methods.

In Section II we review work in the two major themes that relate to this study, namely time series discretization methods and sequential rule mining methods. We give a formal definition of the research problem in Section III. The framework for sequential rule mining and associated methods is presented in Section IV. Section V presents the experimental design used to study the effects of different discretization methods and different settings for key parameters such as sliding window size and aggregation size on the quality of the rules produced. Finally in Section VI some concluding remarks are made and some directions for future research are discussed.

II. RELATED WORK

One of the most widely used discretization methods for time series is the Symbolic Aggregate approXimation (SAX) proposed by Lin et al [6]. The SAX method uses the Gaussian distribution to discretize data into bins that contain values that occur with equal probability.

In [7], a clustering technique is used to transform the time series into symbols representing the geometrical shape of the time-series. In [13], a clustering method based on the concept of Partial K-Completeness and Interestingness is used on Hydrological data.

In terms of temporal rule extraction, Das et al published one of the very first studies on extracting rules from time-series data, which uses a clustering method to discretize the time-series based on geometrical shape and proposed a modified Apriori algorithm to discover rules from a set of discretized time-series [7]. Last et al proposed a general methodology for knowledge discovery from time-series producing fuzzified rules based on an information-theoretic and connectionist approach [8].

Aside from these studies, research in time-series rule mining has revolved mostly around improvement to certain aspects of the aforementioned techniques, such as the discretization step, and different applications of it. For example, Mörchen and Ultsch proposed a new quality score to measure unsupervised discretization of time series, by taking the temporal information into account and searching for persistence, argued as more suitable for knowledge discovery purposes, and offered a discretization algorithm called Persist [9]. Pradhan and Prabhakaran mined useful rules in multi-attribute medical data, specifically from multiple surface electromyogram (EMG) data to analyze muscle movement behaviors with sequential apriori algorithm [10]. Rule mining as a part of an integrated time-series data mining of medical therapy data as part of a hospital information system has also been explored by Abe et al [11]. Temporal rule mining was also used by Warasup and Nukoolkit [12] who proposed the usage of symbolic aggregate approximation (SAX) as the discretization technique for financial data analysis.

III. Problem Definition

The temporal rule extraction problem in general can be stated as the discovery of rules that associate the occurrence of an event of interest B within a given time period T of the occurrence of another event A. The events A and B are represented by items or sets of items (henceforth referred to as itemsets). We will first formally define the notion of itemsets in the context of temporal rule extraction.

Given a set of n time series variables: $X_1, X_2, ..., X_n$ that are considered to be predictors of another time series Y, we first obtain the discretized versions of the predictor variables as sets $D_1, D_2, .., D_i, .., D_n$ respectively, where each D_i is itself a set of symbols obtained by discretizing variable X_i. An itemset I can now be defined as:

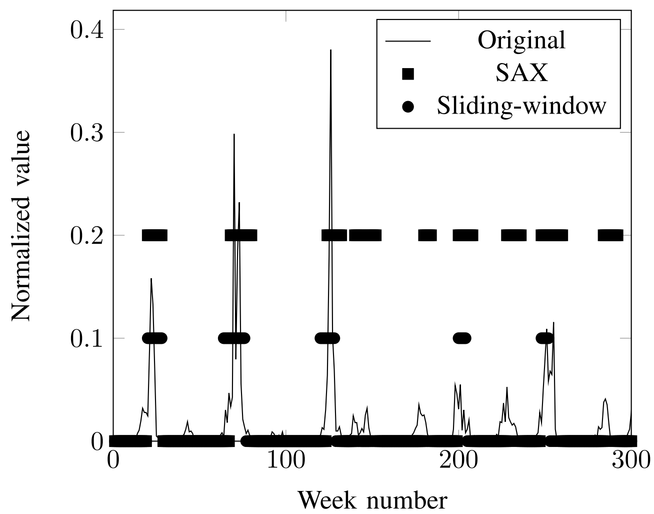
$$I \subseteq DS \tag{1}$$

where $DS = \bigcup_{s=1,n} D_s$I.

Thus an itemset is essentially a set of co-occurring items as in classical association rule mining, but with the added constraint that their occurrence is sequential in nature.

A temporal rule spanning a time period T is denoted by $(A \overset{T}{\Rightarrow} B)$ where A,B are itemsets, $supp(A) >$ *minsup*, $supp(B) >$ *minsup* represent the support of itemsets A and B respectively, while *minsup* is a user defined minimum support theshold; confidence of the rule $c(A \overset{T}{\Rightarrow} B) = \frac{supp(A,B,T)}{supp(A)} >$ *minconf*, a user defined threshold on confidence, and T is a user defined time horizon that specifies that itemset B occurs at most T units of time after the occurrence of itemset A.

Fig. 1: Discretization result comparison. Values of 0.1 and 0.2 have been selected to signify which part of the time-series signal is discretized as a high-signal by SAX and our proposed sliding window approach.



In the context of the aphids dataset we restrict the itemset B to strings containing the symbol that denotes high occurrence of aphid count as the focus of the research is to discover events that lead to high levels of aphid infestation.

IV. A Generic Framework for Temporal Rule Mining from Multiple Time Series

This section will elaborate in detail on the framework that we used to mine temporal rules. Three major steps are involved, namely discretization of time series variables, rule extraction and finally, rule evaluation.

A. *Time-series discretization*

Since most of sequential rule mining algorithms work on data in the form of strings of symbols, one of the most crucial steps in this framework is to find the most appropriate way to transform numerical values of the time-series into symbolic strings. Different strategies have been proposed by various researchers to suit the needs of the data they worked on.

In this research, we use the simple but versatile Symbolic Aggregate approXimation (SAX) proposed by Lin et al for all but one of the time series variables. The result with SAX on ecological species count observation series, which follows a Poisson distribution, is not satisfactory. When faced with such a problem the usual procedure is to log-transform the numerical value to fit the Gaussian distribution, but such transformation with ecological count data for the purpose of satisfying the parametric assumption should be avoided [14]. This is because ecological species count data is very often sparse, containing many zero values and as such a log transformation is not suitable. Fig. 1 shows that discretizing the time-series into a low-high two-symbol string with SAX will incorrectly assign many low-valued peaks to the high symbol.

In the case of such time-series data, we suggest that it is better to use a simple sliding window algorithm to segment the series into two-symbol strings, which represents low and high values. The sliding window algorithm that we propose works by having two segments of a particular size starting from the beginning of the series which slide across the series incrementally. A model is built by using training data in the left segment, and the model is then deployed on new unseen data arriving in the right segment. If the root mean square error of the model on the test (right) segment exceeds a certain threshold, then the data element that defines the boundary between the left and right segments is considered to be a cut point. A cut point represents a transition from either a low signal state to a high signal state or vice-versa. In order to distinguish between the two cases we record the average signal value between the left and right segments. If the average of the right segment is larger, then a transition from a low state to a high state is indicated, else the transition occurs in the opposite direction, from high to low.

The comparison made in Fig. 1 shows that our proposed algorithm is more selective in indicating which peaks are considered to represent high occurrences of aphids. Moreover, the change-detecting nature of the algorithm means that the segmentation is not made according to the absolute value in the time series, but the changes in the value which indicate concept changes. This explains why the high symbol generated by the sliding window algorithm seems to be segmenting a little bit ahead of the actual peak. This behavior is expected to be useful in detecting a pest outbreak, where the identification of when a concept change happens is more important than identifying when a peak occurs. Although there is no rigid objective criterion as to which discretization is better, the effectiveness of the discretization can be indirectly evaluated by looking at the rules produced in the consequent steps.

The pseudo-code for this sliding window based discretization algorithm is illustrated in Algorithm 1. The algorithm is intuitive and has been explored before as a way to segment time-series along with variations in the selection of the regression model (linear vs non-linear) to fit the data and ways to measure the error [17]. This generic algorithm can be implemented by utilizing any type of learning scheme. In this research, we have chosen to use a multiple linear regression model in conjunction with a lagged data, 1-step ahead prediction training/testing regime. Regression was a natural choice as the underlying data is numeric in nature and the linear variant is efficient while being reasonably robust in terms of predictive accuracy. In this algorithm, one important parameter that has to be tuned is the ϵ, which will signify the sensitivity of the change detection. This parameter can be optimized by incrementing the value until there is no change in the discretization result, which suggests that the detected changes are significant enough.

B. Rule Extraction

As introduced in the Problem Definition section of this paper, we used an association rule mining algorithm which produces rules in the form:

if A occurs, then B occurs within time T.

where A,B are itemsets. Instead of representing symbols like in [7], the A and B represent subsequences. If the above rule is denoted as $A \overset{T}{\Rightarrow} B$, then we calculate the confidence of each rule as:

Algorithm 1 Sliding-window model based discretization

Require: D, block-size, ϵ (threshold for change detection)
$a \leftarrow 1$
$i \leftarrow 1$
while not finished scanning the time-series **do**
  $b \leftarrow a$+block-size
  model $\leftarrow$ build-model($D[a{:}b]$)
  error $\leftarrow$ test-model($D[b{:}b$+block-size])
  if error $>$ ϵ $\wedge$ avg(D[a:b])$<avg$(D[$b{:}b$+block-size]) **then**
    highpoint[i] $\leftarrow b$
    $i \leftarrow i+1$
  end if
  $a \leftarrow a+1$
end while
return highpoint

$$c(A \overset{T}{\Rightarrow} B) = \frac{supp(A,B,T)}{supp(A)}, \tag{2}$$

where

$$supp(A,B,T) = \left|\{i \mid a_i = A \wedge B \in \{a_{i+1},...,a_{i+T-1}\}\}\right| \tag{3}$$

and a_i is the symbol that occurs in the i^{th} time step.

Equation (3) represents the number of occurrences of A that are followed by B within a given time period T. The pseudocode of a generic implementation of this technique is presented in Algorithm 2 and can be optimized or modified in various ways to suit any need.

C. Rule Selection criteria

One issue with association rule mining in general and sequential rule mining in particular is that a large amount of rules may be generated, most of which may be trivial and/or uninteresting. Thus, the selection of those rules that are significant and interesting is a challenging task. Adopting the idea of support and confidence from associative rule mining could be useful. Confidence and support are the two most commonly used metrics for measuring rule quality, but a number of researchers have devised other measurements of interestingness for association rules such as the J-measure and Mutual Information, and have subjected these measures to validation and testing [15][16]. The J-measure [19] could be used here and is defined as:

$$\begin{aligned} j(A \overset{T}{\Rightarrow} B) = p(A) * \Big(& p(B|A) \cdot \log\big(\frac{p(B|A)}{p(B)}\big) \\ & +(1-p(B|A)) \cdot \log\big(\frac{1-p(B|A)}{1-p(B)}\big)\Big) \end{aligned} \tag{4}$$

Algorithm 2 Rule-mining algorithm

Require: $D_{1,...,n}$, minsupp, minconf, T
  $I \leftarrow$ generate-itemset(D)
  counter $\leftarrow 0$
  for i=1 to n **do**
    for all a in I_i **do**
      for all b in I_i that occurs in T time steps after a **do**
        confidence $\leftarrow$ support(a,b,T) / support(a)
        if confidence $>$ minconf $\wedge$ support(a) $>$ minsupp **then**
          rules[counter] $\leftarrow a \overset{T}{\Rightarrow} b$
          counter $\leftarrow$ counter + 1
        end if
      end for
    end for
  end for
  return rules

In this context, $p(A)$ is the probability of pattern A occurring among all itemsets of the same length generated from the sequence, while $p(B|A)$ is the probability of pattern B occurring within T time period after the pattern A. The left-hand term gives weight to the frequency of the pattern A, and the right-hand term is the cross-entropy or the information gain. Since there are many rules with high confidence with low support, and vice versa, J-measure is useful because it combines and gives a balanced measurement between the support and confidence. Practically, it can be used as a sound method to create an additional criterion to rank rules.

In this research we consider T as an additional constraint measuring the usefulness of the rules generated. There is obviously little benefit in mining rules with high confidence and support but which span over a long period of time. That is the real world equivalent of saying a plane crash will happen within the next decade. The statement carries a very high level of confidence, but is not particularly useful because of the excessive length of the prediction period.

D. Rule Format Extension

We extend the rule format and algorithm to accommodate multiple antecedents from different time-series in the form:

if A_1 and A_2 and ... and A_h occur within V units of time, then B occurs within T time units.

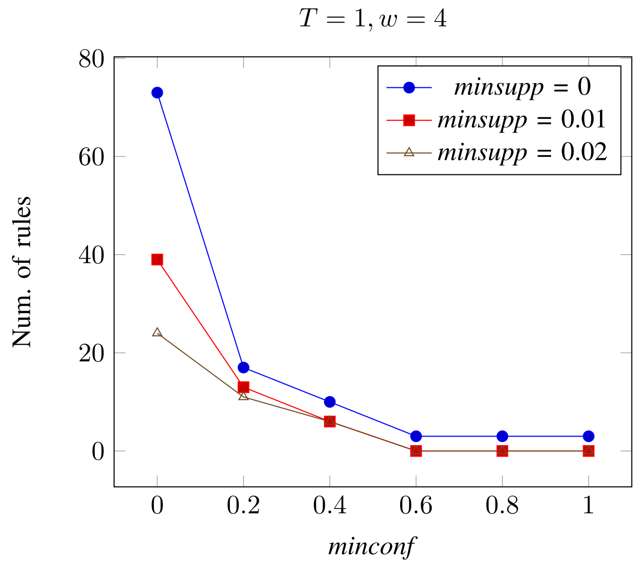
The above rule can be denoted by $A_1 \wedge ... \wedge A_h \overset{V,T}{\Rightarrow} B$. This opens up the possibility of mining from multiple time-series and extracting interactions between the variables involved.

V. Experimental Study

A. Experimental Configuration

The methods described in the previous section will be empirically tested on a dataset which comprised of aphid trap catches recorded by Crop & Food Research, Lincoln, Canterbury, and weekly weather data consisting of 15 weather variables recorded at the Canterbury Agricultural Research Center, Lincoln, New Zealand, spanning over 19 years as described in [2]. In this research, the number of variables used are limited to: Cumulative weekly rainfall, Wind run (km/day), Average air temperature, Potential deficit (accumulated excess of Penman over rainfall), Penman potential evaporation(mm), and Solar radiation (MJ/m2).

Fig. 2: Number of rules generated with different *minconf* and *minsupp*



The experiment focused on whether the proposed framework and methods could discover interesting rules from the multiple time series on aphid infestation, and also to test the sensitivity of parameter values on the results obtained. Since the focus of interest is prediction of high incidence of aphid infestation the rules extracted are restricted to those that feature high aphid count occurrence on rule consequents. We also conduct a sensitivity analysis on key parameters such as window size (w), prediction time horizon (T), minimum support threshold ($minsupp$) and minimum confidence threshold ($minconf$).

All experiments were run on a Core i7 processor configuration running under Windows 7 with Matlab as the main programming tool.

B. Effects of Minimum Support Threshold

Fig. 2 shows the effect on the number of rules generated as the $minsupp$ threshold is increased. As expected, when $minsupp$ increases from 0 a steady decrease in the number of high confidence rules (with confidence ≥ 0.8) is observed. Interestingly, with no constraint on support we observe that a substantial number of rules (numbering 10) with confidence of 1 predict high occurrence of aphid infestation within a week of the triggering event firing on the rule antecedent. This is due to the fact that each time step represents a week as the aphids data was collected on a weekly basis. The same trend is observed for higher $minsupp$ threshold values, although the number of high confidence rules generated reduces by a factor of 2 or more.

Fig. 3: Number of rules generated with different window size of discretization

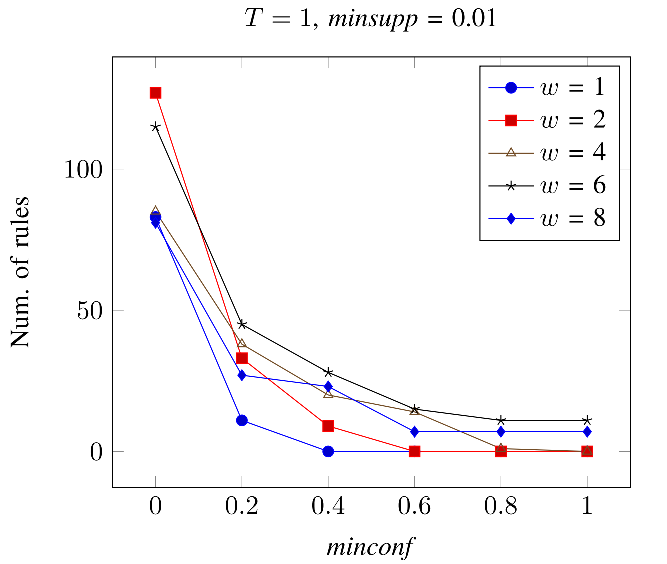
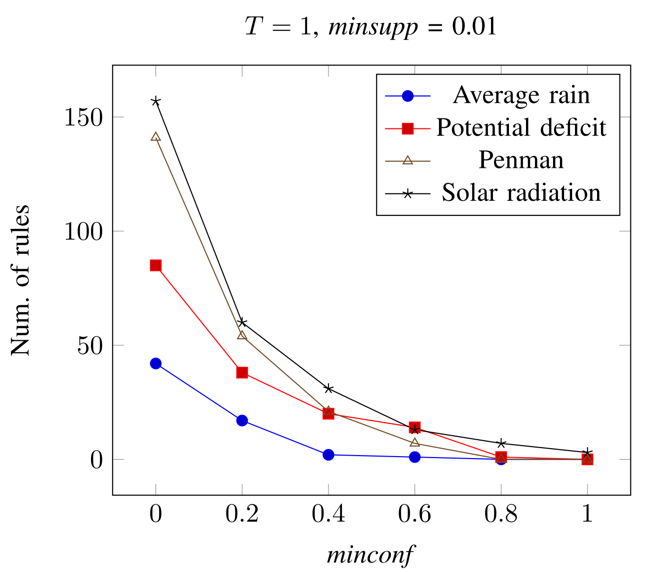


Fig. 4: Number of rules generated by different antecedent variables



C. Effects of Window Size

Fig. 3 shows that as the window size w increases from 1 a higher number of high confidence rules can be obtained. Basically, the reason is that smaller window sizes are vulnerable to the effects of noise. Small window sizes capture smaller transitions between states when compared to larger window sizes, thus effectively identifying smaller peaks. The problem with smaller peaks is that they are associated with random behavior, thus trigger conditions on rule antecedents are rendered ineffective, giving rise to low confidence rules. As the window size increases from 2 the noise level decreases and the number of high confidence rules increases.

TABLE I: Some rules produced by the method

w	Antecedent	Rule	Supp. (%)	Conf. (%)	J-Measure	Fig.
4	Potential deficit	ddddaaa $\overset{1}{\Rightarrow}$ b	8.86	70	0.0036	5
4	Penman evaporation	aaaabcddd $\overset{2}{\Rightarrow}$ b	3.52	75	0.0024	6
4	Potential deficit & Mean temperature	aaaaabbb, dddcb $\overset{4,1}{\Rightarrow}$ b	0.8	100	0.0009	7

However, increasing the window size from 6 to 8 results in a reduction in the number of high confidence rules. Beyond a certain threshold on window size, dependent on the nature of the underlying dataset, some high valued peaks will not be detected and thus some rules, including some with high confidence rules will not be generated. Moreover, too large a window size is undesirable because the period of time that T represents corresponds to the window size. Having a wide window size for T means that the rules produced have a longer window of prediction, rendering them less useful. We decided to use $w = 4$ as the window size, which is a trade-off between the number of rules generated and the resolution of prediction.

D. Effects of Rule Antecedent Variable

A visualization of the difference in the ability of the variables to produce rules with varying levels of confidence is shown in Fig 4. The number of high confidence rules which a variable can produce is an indicator of its relative importance in influencing a high aphid count outcome. In this context, we can infer that the cumulative rainfall is relatively less important than the other variables, and on the other hand, the Penman potential evaporation seems to be a very strong feature, being able to produce very high confidence rules.

Table 1 shows some of the rules that are discovered using the algorithm. These rules are visualized in Fig. 5, 6, and 7, which show the occurrence of high aphid count following the triggering events which are captured by the antecedent of the rule. The string symbols used as the antecedent and the consequence of the rules are from the discretized time series with 4 levels of intensity, the character a and d for the lowest and the highest values respectively. Thus Fig. 5 shows that 4 consecutive occurrences of high potential deficit values (denoted by symbol "d") followed by 3 consecutive low level occurrences of potential deficit (denoted by symbol "a") triggers a high aphid count. This can be interpreted as the occurence of a rapid drop after a period of high values in the antecedent. The high confidence of the rule is evident from the visualization that shows that the antecedent pattern is followed almost always by a peak in the aphid count value. The support of the rule is also evident from the number of co-occurrences of the antecedent and consequent patterns - i.e. the number of times that the string *dddaaab* occurs in the data.

Likewise, Fig. 6, and 7 visually show the confidence of rules: $aaaabcddd \overset{2}{\Rightarrow} b$ and $aaaaabbb, dddcb \overset{4,1}{\Rightarrow} b$ respectively. The latter rule clearly shows the effect of having multiple variables in the rule antecedent. The inclusion of Mean temperature with Potential deficit in the rule antecedent

Fig. 5: Rule visualization: ddddaaa $\overset{1}{\Rightarrow} b$

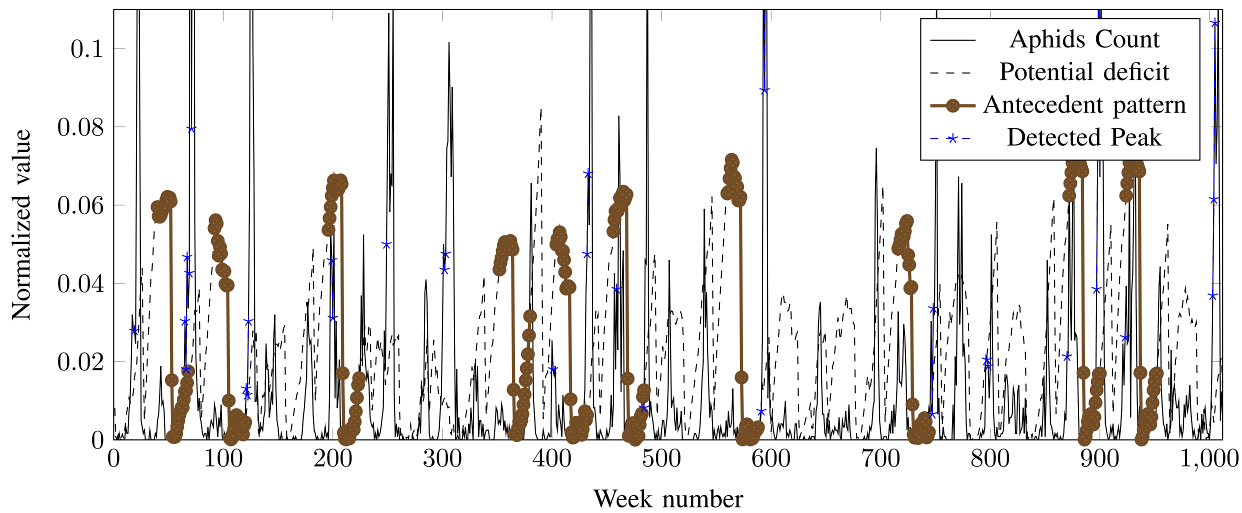


Fig. 6: Rule visualization: aaaabcddd $\overset{2}{\Rightarrow} b$

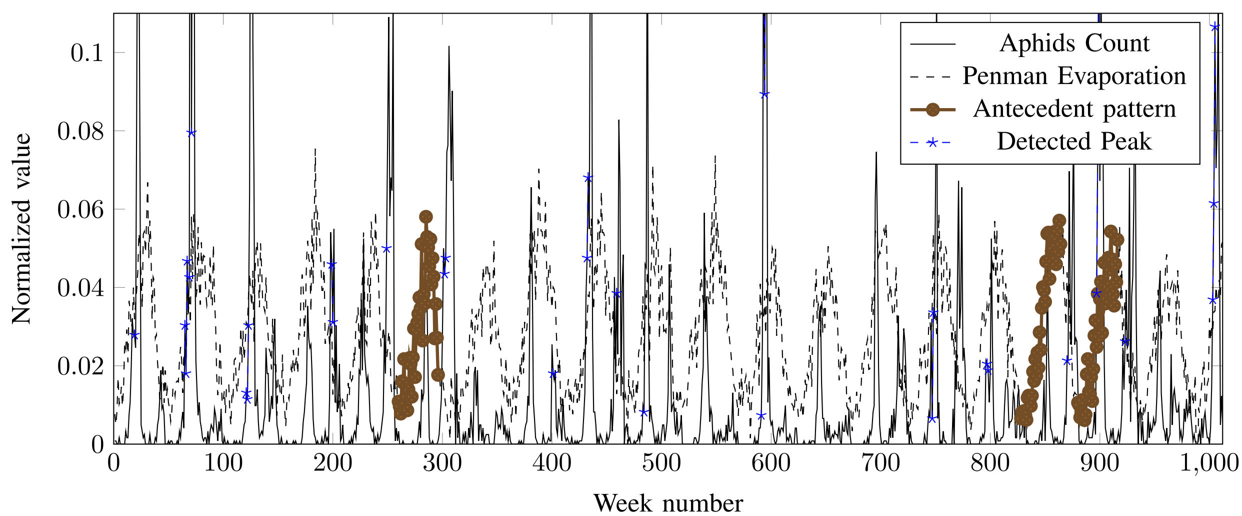
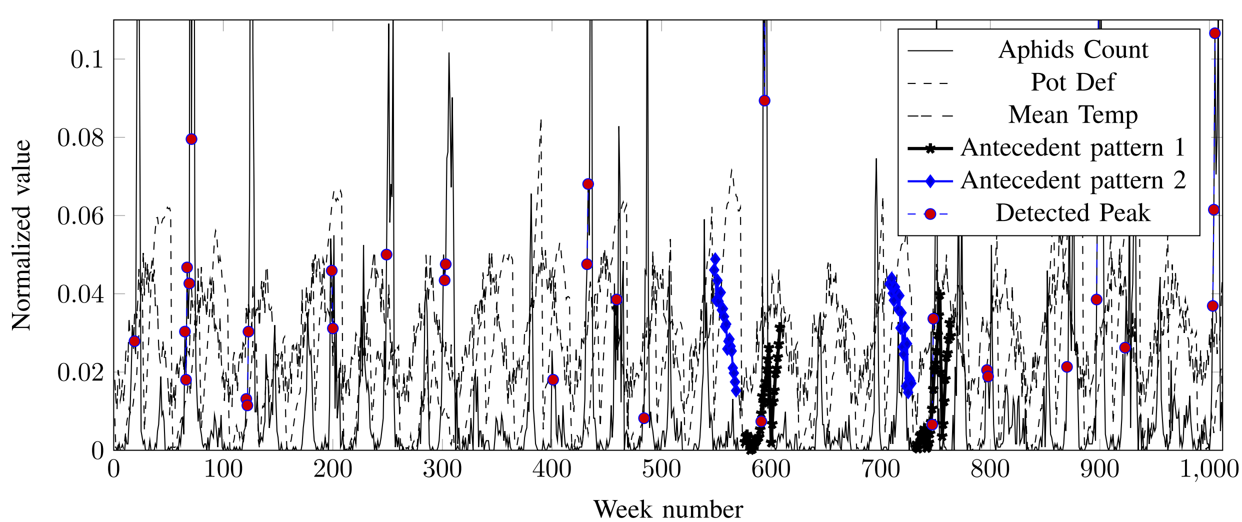


Fig. 7: Rule visualization: aaaaabbb, dddcb $\overset{4,1}{\Rightarrow} b$



results in an increase in confidence of 30% over the use of Potential deficit alone. However, the trade-off caused by the inclusion of the additional variable has caused the support and J measure metrics to decrease substantially. It is also interesting to note that the Potential deficit signature(row 3 of Table 1) is completely different from that of its signature in the first rule (row 1 of Table 1).

Overall, it is evident that the rules generated are useful in identifying high levels of aphid infestation. For example, rules 1 and 2 in Table 1 identify trends in Potential deficit and Penman evaporation variables respectively that lead to high aphid count with reasonably high levels of confidence (70% and 75% respectively), while giving growers an adequate time periods (1 week and 2 weeks, respectively) to implement suitable pest control measures.

VI. Conclusions & Future Work

In this paper, a generic framework for mining temporal rules from multiple time-series has been defined and a case study on an ecological dataset has been demonstrated. The methods described have been shown to be able to extract some useful temporal rules and have the potential to be applied in many other fields in which such rules could be used to improve the ability of humans to predict the likelihood of an incident happening based on currently available observations.

This work could be extended in various ways. The use of fuzzy representations and rules is promising, since it reflects better how the rules are represented in human language and concepts, as has been done in [18]. Building a classifier which is able to employ the mined rules to improve long-term prediction is also a possible extension. Decomposing the time-series into trend and seasonal decomposition is also a promising way to pre-process the data, as it could better reveal the interactions between the variables. Rule extension to accommodate spatio-temporal information is also of interest, as ecological modeling often involves both spatial and temporal dimensions.

Acknowledgment

We would like to thank The New Zealand Institute for Plant & Food Research for providing the aphids dataset. This study is also made possible by a scholarship from the New Zealand Ministry of Foreign Affairs and Trade under the NZ ASEAN Scholar Award scheme. Special thanks also goes to fellow researchers at the Knowledge Engineering and Discovery Research Institute (KEDRI) at AUT, Bio-Protection Research Centre at Lincoln University, and the reviewers for the inputs and discussions.

References

[1] T. C. Fu, "A review on time series data mining," *Applicat. of Artificial Intell*, vol. 24, no. 1, pp. 164-181, 1991.

[2] S. P. Worner *et al.*, "Improving prediction of aphid flights by temporal analysis of input data for an artificial neural network," *New Zealand Plant Protection*, pp. 312-316, 2002.

[3] M. J. Watts and S. P. Worner, "Using artificial neural networks to determine the relative contribution of abiotic factors influencing the establishment of insect pest species," *Ecological Informatics*, vol. 3, no. 1, pp. 64-74, 2008.

[4] G. Lankin-Vega *et al.*, "An ensemble model for predicting Rhopalosiphum padi abundance," *Entomologia Experimentalis et Applicata*, vol. 129, no. 3, pp. 308-215, 2008.

[5] M. J. Watts and S. P. Worner, "Using MLP to Determine Abiotic Factors Influencing the Establishment of Insect Pest Species," *Int. Joint Conf. Neural Networks*, Vancouver, BC, 2006, pp. 1840-1845.

[6] J. Lin *et al.*, "A symbolic representation of time series, with implications for streaming algorithms," *Proc. 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*, San Diego, CA, 2003, pp. 2-11.

[7] G. Das *et al.*, "Rule Discovery from Time Series," *Proc. 4th Int. Conf. Knowledge Discovery and Data Mining*, New York City, NY, 1998, pp. 16-22.

[8] M. Last *et al.*, "Knowledge discovery in time series databases," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 31, pp. 160-169, Feb. 2001.

[9] F. Mörchen and A. Ultsch, "Optimizing time series discretization for knowledge discovery," *Proc. 11th ACM SIGKDD Int. Conf. Knowledge Discovery in Data Mining*, Chicago, IL, 2005, pp. 660-665.

[10] G. N. Pradhan and B. Prabhakaran, "Association rule mining in multiple, multidimensional time series medical data," *IEEE Int. Conf. Multimedia and Expo*, New York, NY, 2009, pp. 1720-1723.

[11] H. Abe *et al.*, "Implementing an Integrated Time-Series Data Mining Environment Based on Temporal Pattern Extraction Methods: A Case Study of an Interferon Therapy Risk Mining for Chronic Hepatitis," *New Frontiers in Artificial Intelligence*, vol. 4012, pp. 425-435, 2006.

[12] K. Warasup and C. Nukoolkit, "Discovery association rules in time series data," *KMUTT Research and Development Journal*, vol. 29, no. 4, pp. 447-462, 2012.

[13] D. Wan *et al.*, "Discovery association rules in time series of hydrology," *IEEE Int. Conf. Integration Technology*, Shenzhen, China, 2007, pp. 653-657.

[14] R. B. O'Hara and D. J. Kotze, "Do not log-transform count data," *Methods in Ecology and Evolution*, vol. 1, no. 2, pp. 118-122, 2010.

[15] S. Lallich *et al.*, "Association rule interestingness: measure and statistical validation," *Quality measures in data mining*, vol. 43, pp. 251-275, 2007.

[16] P. N. Tan *et al.*, "Selecting the right interestingness measure for association patterns," *Proc. 8th ACM SIGKDD Int. Conf. Knowledge Discovery in Data Mining*, Edmonton, AB, 2002, pp. 32-41.

[17] E. Keogh *et al.*, "Segmenting time series: A survey and novel approach," *Data mining in time series databases*, vol. 57, pp. 1-22, 2004.

[18] C. H. Chen *et al.*, "Fuzzy data mining for time-series data," *Applied Soft Computing*, vol. 12, no. 1, pp. 536–542, 2012.

[19] P. Smyth and R. M. Goodman, "An information theoretic approach to rule induction from databases," *IEEE Trans. Knowl. Data Eng.*, vol. 4, pp. 301-316, Aug. 1992.