The First IEEE International Conference on Computer Communication and the Internet (ICCCI 2016)

Sensor Time Series Association Rule Discovery Based on Modified Discretization Method

Ruidong Xue[1], Tingting Zhang[2], Dehua Chen[1], Jiajin Le[1], Mehrzad Lavassani[2]
[1]School of Computer Science and Technology, [2]Department of Information Technology and Media
[1]Donghua University, [2]Mid Sweden University
[1]Shanghai, P.R.China, [2]Sundsvall, Sweden
RuidongXue@hotmail.com, tingting.zhang@miun.se, chendehua@dhu.edu.cn, lejiajin@dhu.edu.cn, Mehrzad.Lavassani@miun.se

***Abstract*—Association rule discovery from sensor time series is a challenge. Because the time series has high dimensional, numerical and continuous nature. However the general association methods can only deal with data which are symbolic and discrete. And the general association methods have high processing time consumption when the data have high dimension.**

So a useful framework is proposed, which is pre-processing, representation, discretization and temporal association mining. In the discretization section, a modified discretization method is proposed which can combine the advantages of other methods, such as piecewise aggregate approximation (PAA), knee point selection, symbolic aggregate approximation (SAX) and monotonicity feature extraction. In the association section, a modified Apriori algorithm is proposed to discover special patterns and normal rules.

Keywords—sensor time series; discretization method; temporal association method

I. INTRODUCTION

In recent years, more and more time series are generated from sensors. The time series are different from general data set [1]. Thus it is a challenge to discover association rules from sensor time series. To discover association rules from sensor time series, several problems have to be solved. First is high dimension of the time series. Second is current association methods which can only deal with symbolic data, however the time series are numeric data. Third is current association methods do not consider the order of the items, however the time series have the temporal nature [2].

To the problem of large scale, representation method can reduce the large scale and extract the key information from the time series. To the problem that the current association methods can only deal with symbolic value, however the time series are numeric value. Suitable discretization method can transfer the numeric value to symbolic value [3].

In this paper, a modified discretization method is proposed, which can combine the advantages of two representation methods and two discretization methods. The two representation methods are PAA and knee point extraction method. The two discretization methods are SAX and monotonicity feature extraction method.

Supported by Shanghai Science and Technology Action Planning Projects (*15511106900).*
Supported by Shanghai Science and Technology Commission Basic Research Projects *(16JC1400802)*

To the problem that the current association methods never think about the order of the items, however the time series have temporal nature. This paper modify the Apriori algorithm to discover special patterns and normal rules from multiple time series.

The contribution of this paper is first to propose a framework to discover potential knowledge from sensor time series, which is pre-processing, representation, discretization and temporal association mining. Second a modified discretization method is proposed which can combine the advantages of other representation methods and discretization methods. Third is the modification of the Apriori algorithm, which can discover special patterns and normal rules from the time series.

Section Ⅱ describes the theory and related work of this paper. Section Ⅲ introduces the implementation of the study. Section Ⅳ describes the meaning of the temporal association rules. And the evaluation of the result. Section Ⅴ is the conclusion of the entire study.

II. THEORY AND RELATED WORK

In this section, several popular methods are introduced, which are min-max normalization, z-score normalization, PAA, knee point selection [4], SAX, monotonicity feature extraction method and temporal association method.

A. Min-max Normalization

Min-max normalization finds the minimum value and the maximum value of the data set. After min-max normalization, the new value is between 0 and 1. The variable a is original variable, a' is new variable, min_a is the minimum value of the original data set, max_a is the maximum value of the original data set. The formulation is shown below.

$$a' = \frac{a - \min_a}{\max_a - \min_a} \quad (1)$$

B. Z-score Normalization

For z-score normalization, the original value is normalized through the mean value and the standard deviation. The original value called v, the mean value called u and the standard deviation called δ. The formulation is shown below.

$$v' = \frac{v-u}{\delta} \quad (2)$$

C. Piecewise Aggregate Approximation (PAA)

The time series is defined as $X = x_1, \ldots, x_n$. Let N to be the length of the segmentation. The range of N is from 1 to n. Let n to be the length of original time series. The formulation is shown below [5].

$$\bar{x}_i = \frac{N}{n} \sum_{j=n/N(i-1)+1}^{(n/N)i} x_j \quad (3)$$

This formulation can compress the time series from n dimensions to N dimensions. The whole time series is split into N equal length pieces. The process is shown in Fig. 1.

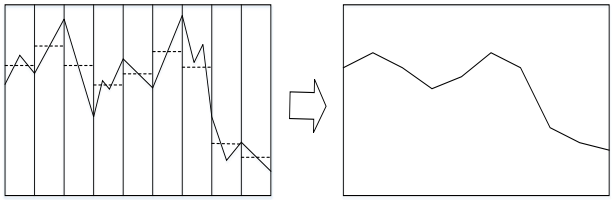

Fig. 1. PAA.

D. Knee Point Selection

The knee point is the turning points of the time series. The knee point is bigger than left N points and right N points, or smaller than left N points and right N points [6]. The example of knee point is shown in Fig. 2.

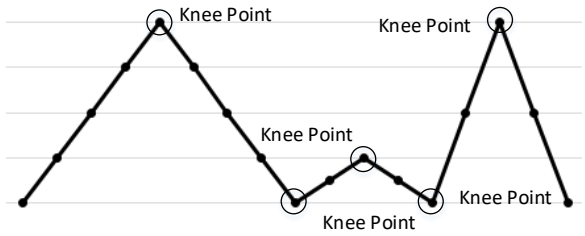

Fig. 2. Knee points.

E. Symbolic Aggregate Approximation (SAX)

Symbolic aggregate approximation (SAX) is a symbolic representation of the time series [7]. The description is shown in Fig. 3.

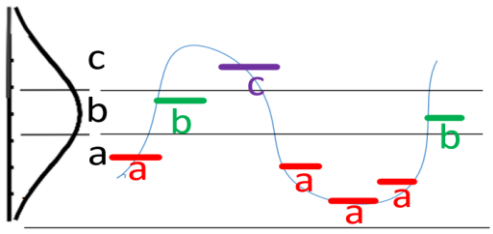

Fig. 3. SAX symbols representation.

The above figure means when the value of the point is under the smallest breakpoint, the point is mapped to the characteristic *a*. When the value bigger than the smallest breakpoint and less than the second smallest breakpoint, the point is mapped to characteristic *b* [8]. When the value bigger than the biggest breakpoint, the point is mapped to characteristic *c*.

F. Monotonicity Feature Extraction

The monotonicity feature extraction method represents the monotonicity of the time series. Three symbols are applied to represent the time series. The description of three symbols are shown in Table I. The threshold is user defined variable.

TABLE I. DESCRIPTION OF THREE SYMBOLS

Symbol	Definition
Up	(Instance[end]-Instance[begin]) / Instance[begin] > Threshold
Down	(Instance[end]-Instance[begin]) / Instance[begin] < -Threshold
Level	|Instance[end]-Instance[begin]| / Instance[begin] ≤ Threshold

G. Temporal Association Methods

The basic rule format is when the item X once happens, the item Y will happens within time T. The item X and item Y are substrings [9]. The temporal association rule is shown in formula 4.

$$X \overset{T}{\Rightarrow} Y \quad (4)$$

The absolute support of the association rule is shown in formula 5. It counts the total number of the rule in the time series.

$$\sup_a \left(X \overset{T}{\Rightarrow} Y \right) = F(X, Y, T) \quad (5)$$

The formula 6 gives the detail description of $F(X,Y,T)$, which is the total number of the rule. T can be seen as the user defined time unit.

$$F(X,Y,T)=|\{i|x_i=X \wedge Y \in \{x_{i+1},\ldots,x_{i+T-1}\}\}| \quad (6)$$

III. IMPLEMENTATION

There are 23 sensors in this study. These sensors generate the data per minute. The data set is stored in EXCEL file. JAVA language is applied to implement all methods. The temporal association rules are stored in TXT file.

The whole implementation includes the analysis of the time series, data pre-process, representation, discretization and multiple time series association method. Fig. 4 describes the structure of the implementation. The red methods are current popular methods. After comparison and analysis, the modified discretization method is proposed. And the modified Apriori algorithm is proposed. These two modified methods are yellow color in Fig. 4.

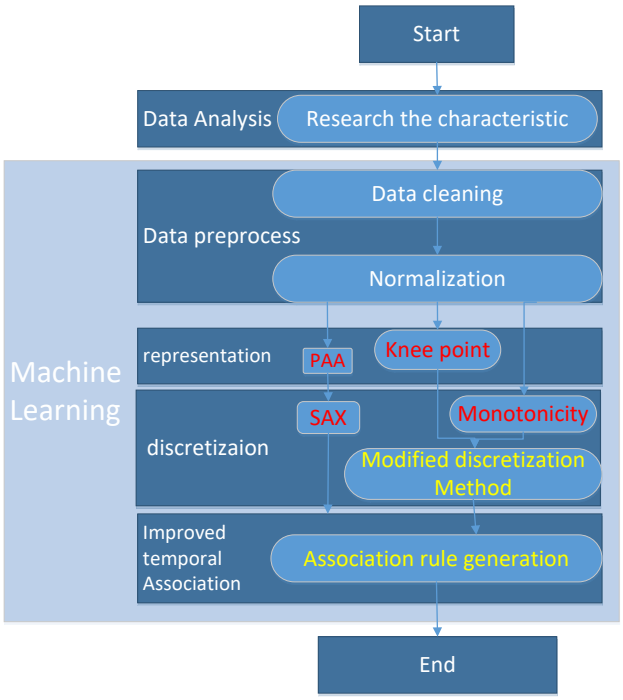

Fig. 4. Structure of the implementation.

A. Data Set Analysis

There are 23 sensors deployed on different parts of the industrial machine. These parts include motors, coolers, pumps, drives and tanks. These sensors record the temperature, pressure, speed or current every minute.

B. Data Pre-process

1) Min-max Normalization

The min-max normalization can make the range of all attributes from 0 to 1. It is convenient to compare multiple time series.

2) Z-score Normalization

Because the SAX needs to analyze the distribution of the time series. So after z-score normalization, the mean value of every time series is 0, and the standard deviation is 1. The purpose of z-score normalization is to help SAX to discretize the time series.

C. Two Representation Methods

After pre-process, the first thing is to reduce the large scale of time series. Two representation methods are tested, which are PAA and knee point selection.

1) Piecewise Aggregate Approximation (PAA)

Every time series is divided to *N* equal-sized segments. We use the mean value of every segment to consist the representation of time series. The formulation of PAA is shown in formula 3.

2) Knee Point Selection

The knee point selection method can reduce the scale of the time series and hold the key information of the time series. The implementation of the knee point selection is shown in algorithm Ⅰ.

Algorithm 1 Knee Points Selection Algorithm

Input: Timeseries: A={a_1, a_2, a_3, a_4, ...}
WindowSize: N
Output: Control Points

```
while(A not end )
do
    while(j<=N)
    do
        if(a_i bigger than left point and right point)
            if(a_i >= a_{i+j} and a_i >= a_{i-j})
                a_i is the important point "peak"
            end if
        end if
        if(a_i smaller than left point and right point)
            if(a_i <= a_{i+j} and a_i <= a_{i-j})
                a_i is the important point "trough"
            end if
        end if
    end while
end while
```

D. Two Discretization Methods

After representation, the scale of the time series is compressed. So it can help to reduce the time cost of the temporal association method. However the association method can not deal with the numeric value. Then the time series need to be transformed from numeric value to symbolic value. So two discretization methods are tested, which are monotonicity feature extraction method and SAX.

1) Monotonicity Feature Extraction

The time span of this method can be defined as daily scale or hourly scale. The description of how to transform the numeric value to symbolic value is shown in Table I.

2) Symbolic Aggregate Approximation (SAX)

The purpose of SAX is to produce symbols with equal probability [10]. Because after z-score normalization, the mean value of time series is 0 and the deviation is 1. So SAX can use breakpoint to consist *k* same area regions which under Gaussian distribution. Variable *k* is user defined value.

However SAX has a problem, which is the assumption that the time series are Gaussian distribution after z-score normalization. After z-score normalization, the distribution of the 23 sensor time series are shown in Fig. 5.

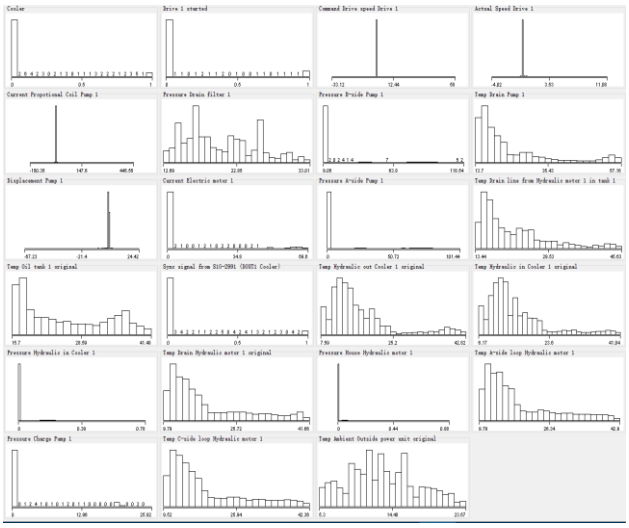

Fig. 5. Distribution of the 23 time series.

According to the distribution of 23 time series, we can conclude that most sensor time series are not Gaussian distribution. So the SAX are not appropriate for this study. To solve this problem, a modified discretization method is proposed in next section.

E. Modified discretization Method

From now there are 2 representation methods and 2 discretization methods are implemented, which are PAA, knee point extraction, SAX and monotonicity feature extraction. However according to the experiment, these methods have some problems. The PAA can impact the result of the z-score normalization and miss some important points [11]. The knee point extraction method can not make every time series has same length, because every time series has different number of the knee points. SAX only represents the big or small characteristic of the time series, it is not enough in this study. And monotonicity feature extraction method can not represent the curve information of the time series.

To avoid disadvantages of these representation and discretization methods, and combine the advantages of them, a modified discretization method is proposed. The modified discretization method uses *up*, *down*, *level*, *peak* and *trough* symbols to represent the time series. These symbols can extract curve information and monotonicity of the time series. The meaning of these symbols is when the value of the time series increase in a substring, we use *up* to represent this substring. When the value of the time series decrease in a substring, we use *down* to represent this substring. When the value of the time series keeps stable in a substring, we use *level* to represent this substring. When there is a point bigger than it is left and right points in a substring, we use *peak* to represent this substring. When the there is a point smaller than it is left and right points in a substring, we use *trough* to represent this substring. The description of these five symbols is shown in Table Ⅱ.

TABLE II. DESCRIPTION OF FIVE SYMBOLS

Symbol	Description	Simplify
Up	(Instance[end]-Instance[begin]) / Instance[begin] > Threshold	u
Down	(Instance[end]-Instance[begin]) / Instance[begin] < -Threshold	d
Level	|Instance[end]-Instance[begin]| / Instance[begin] ≤ Threshold	l
Peak	begin ≤ m ≤ end, Instance[i] ≥ Instance[m]	p
Trough	begin ≤ m ≤ end, Instance[i] ≤ Instance[m]	t

To make these five symbols more clearly, the Fig. 6 gives an example to show how 5 symbols represent the time series.

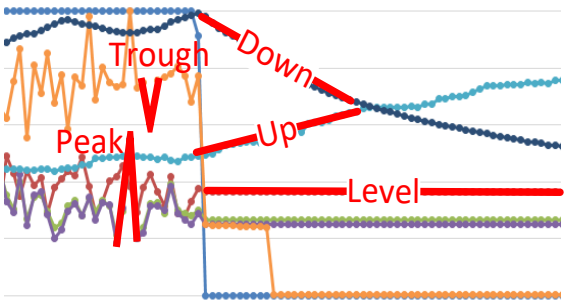

Fig. 6. Example of these symbols.

According to the purpose of this study, the discretization method has to reduce the large scale of the time series, extract monotonicity and curve information of the time series. And these time series have to have same length after discretization. To show the advantage of this modified discretization method, the comparison of modified discretization method and other methods are shown in Table Ⅲ.

F. Multiple Time Series Association Method

Apriori algorithm can only deal with unordered items, however the time series have temporal nature. So a modified Apriori algorithm is proposed in this study.

1) Multiple Time Series Association Rule Mining

TABLE Ⅲ. COMPARISON OF THESE REPRESENTATION AND DISCRETIZATION METHOD

Name	Number reduce	Monotonicity	change point	same point number	Suitable Symbol	Express Symbol
New discretization method	**√**	**√**	**√**	**√**	**√**	**level, up, down, peak, trough**
Improved SAX	√	×	×	√	√	a,b,c
Knee Point Extraction	√	×	√	×	×	Knee Point
Monotonicity Feature Extraction	√	√	×	√	√	level, up, down

There are 23 sensors in this study. So for the rules between two sensors like A $\xrightarrow{T}$ B, which has A_{23}^{2} = 506 permutations. And for the rules in three sensors like A, B $\xrightarrow{T}$ C, which has A_{23}^{3} = 10626 permutations. The rule generation between two time series like A $\xrightarrow{T}$ B is shown algorithm Ⅱ.

Algorithm 2 A $\xrightarrow{T}$ B Rule Generation Algorithm

Input: Time Series Set: D; Time series ID: i, j;
Window size: T
Output: Association Rule

```
leftItemIterator = maxFrequency ( D[i] )
rightItemIterator = maxFrequency ( D[j] )
while( leftItemIterator has value )
do
    leftItem = leftItemIterator.next()
    while ( rightItemIterator has value )
    do
        rightItem = rightItemIterator.next()
        if ( Support ( leftItem -T-> rightItem ) )
            Rule.add ( leftItem -T-> rightItem )
        end if
    end while
end while
```

The implementation of the rules in three time series like A, B $\xrightarrow{T}$ C is shown in algorithm Ⅲ.

Algorithm 3 A, B $\xrightarrow{T}$ C Rule Generation Algorithm

Input: Time Series Set: D; Time series ID: i, j. k
Output: Association Rule

```
leftItemIterator1 = maxFrequency (D[i])
leftItemIterator2 = maxFrequency (D[j])
rightItemIterator = maxFrequency (D[k])
while (leftItemIterator1 has value) do
    leftItem1 = leftItemIterator1.next()
    while ( leftItemIterator2 has value ) do
        leftItem2 = leftItemIterator2.next()
        while ( rightItemIterator has value ) do
            rightItem = rightItemIterator.next()
            if (Support(leftItem1, leftItem2 -T-> rightItem)>minSupportNumber
        and Confidence (leftItem1, leftItem2 -T-> rightItem)>minConfidence)
                Rule.add (leftItem1, leftItem2 -T-> rightItem )
            end if
        end while
    end while
end while
```

2) Frequency Threshold and Support Threshold

After implementing multiple time series association method, the normal rules can be obtained. However it is not enough. The normal rules can only show superficial knowledge of the time series.

Thus to discover potential knowledge of the time series, the *frequency threshold and the support threshold* method is proposed. These two thresholds are user defined variable. The frequency threshold determines whether the item needs to be selected. The support threshold determines whether the rule needs to be selected.

To discover normal rules, we use *above frequency threshold and above support threshold* method. To discover abnormal patterns from frequent itemset, we use *above frequency threshold and below support threshold* method. To discover abnormal patterns from infrequent itemset, we use *below frequency threshold and below support threshold* method. These three situation is shown in Table Ⅳ.

TABLE IV. THREE KINDS OF PATTERNS

Rule	Frequency Threshold	Support Threshold
Normal	Above	Above
Abnomral	Above	Below
Abnomral	Below	Below

For example, the user wants to discover normal rules. So the frequency threshold is set to 100. The frequency of an item I is 500, it is bigger than 100. So the item I will be selected in the itemset. And the support threshold is set to 100, the support of the rule R is 500. So the support of the rule R is bigger than 100. The rule R will be selected as the normal rule.

IV. RESULT AND EVALUATION

The multiple time series association rules, which can help us to discover potential knowledge and useful information from the multiple time series.

In the discretization section, the discrete time series has five different symbols which are *up*, *down*, *level*, *peak* and *trough*. These five symbols are called *u*, *d*, *l*, *p* and *t*. For example the discrete time series *uuuupdddd* means the value of the time series increases for 4 time units and has a peak for 1 time unit, then the value of the time series decreases for 4 time units.

In this experiment, the frequency threshold is set to 5%, the support threshold is set to 10% and the length of the discrete time series is 4319 after discretization. So the frequency threshold number is 215.95 and the support threshold number is 431.9.

A. Rule Description

The rules have three categories, which are normal pattern, abnormal pattern from frequent itemset and abnormal pattern from infrequent itemset. The result has two kinds of rules, which are rules between two sensors and rules in three sensors. The rules between two sensors like A $\xrightarrow{T}$ B are shown in Table Ⅴ. The rules in three sensors like A, B $\xrightarrow{T}$ C are shown in Table Ⅵ.

TABLE V. RULES BETWEEN TWO ITEMS

Resources	New Discretization Method
Length	4319
Frequency Threshold	5%
Min Confidence	90%
Number of Sensors	23
Suport Threshold	10%
Window Size	15
Classify	Above Frequency Threshold Above Support Threshold
Size	549KB
Operation Time	935590ms
Rule Number	10000
Example	dl -->lllllllllllll
Classify	Above Frequency Threshold Below Support Threshold
Size	930KB
Operation Time	778180ms
Rule Number	20100
Example	td -->lllllllllll
Classify	Below Frequency Threshold Below Support Threshold
Size	286KB
Operation Time	2102033ms
Rule Number	5027
Example	dp -->uuuu

TABLE VI. RULES IN THREE ITEMS

Resources	New Discretization Method
Length	4319
Frequency Threshold	5%
Min Confidence	90%
Number of Sensors	23
Suport Threshold	10%
Window Size	15
Classify	Above Frequency Threshold Above Support Threshold
Size	40802KB
Operation Time	54651748ms
Rule Number	860000
Example	dd , lllllll -->dd
Classify	Above Frequency Threshold Below Support Threshold
Size	129825KB
Operation Time	45137305ms
Rule Number	2660000
Example	tt , lu -->ll
Classify	Below Frequency Threshold Below Support Threshold
Size	30621KB
Operation Time	49624203ms
Rule Number	1210000
Example	uuud , ddu -->up

For the rules in Table Ⅴ and Table Ⅵ, the first class means the frequency of items is bigger than 215.95 and the support number of the rules is bigger than 431.9. These rules can be regarded as the *normal rules* of the time series. The second class means the frequency of the items is bigger than 215.95 however the support number of the rules is smaller than 431.9. These rules can be regarded as *abnormal pattern from frequent itemset*. The third class means the frequency of the items is smaller than 215.95 and the support number of the rules is smaller than 431.9. These rules can be regarded as *abnormal pattern from infrequent itemset*.

B. *Rule Evaluation*

According to the evaluation, we can conclude the temporal association rules have above 90% accuracy. To make the evaluation more clearly and convincingly, randomly showing the rule $pppu \xrightarrow{T} ud$. The support number of this rule is 20 and the confidence of this rule is 90%. For original time series, several rules are shown in Fig. 7. 20 patterns are marked by the blue dots. Because the limitation of the size, 4 rules are shown in the figure.

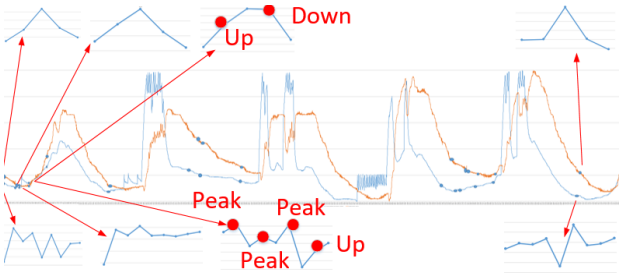

Fig. 7. Evaluation of the patterns in the original time series.

V. CONCLUSION

Because the time series have large size, numeric and temporal nature. However the characteristic of the current association methods is high time cost, and can only deal with symbolic value and unordered items. This confliction brings a challenge to discover useful knowledge from sensor time series. Through analysis and experiment, this paper propose a framework to overcome this challenge. The framework is pre-process, representation, discretization and temporal association mining.

In the implementation section, two representation methods and two discretization methods are tested. The experiment shows that these methods are not appropriate for this study. So a modified discretization method is proposed which can combine the advantages of these methods and avoid the disadvantages of them.

The modified Apriori algorithm considers the temporal nature of the sensor time series. Furthermore, in order to discover the potential knowledge from the sensor time series, the frequency threshold and the support threshold method is proposed in this paper.

REFERENCE

[1] Varun Chandola1, Olufemi A. Omitaomu1, Auroop R. Ganguly1, Ranga R. Vatsavai1, Nitesh V. Chawla2, Joao Gama3, Mohamed M. Gaber4 "Knowledge Discovery from Sensor Data (SensorKDD)".

[2] Tak-chung Fu, " A review on time series data mining", Engineering Applications of Artificial Intelligence 24 (2011) 164–181

[3] Tim Schluter and Stefan Conrad, " About the Analysis of Time Series with Temporal Association Rule Mining", 978-1-4244-9927-4/11/$26.00 ©2011 IEEE

[4] Eamonn Keogh, Selina Chu, David Hart, Michael Pazzani, "Segmenting Time Series: A Survey and Novel Approach"

[5] Eamonn Keogh1, Kaushik Chakrabarti2, Michael Pazzani1 and Sharad Mehrotra, 'Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases', Knowledge and Information Systems (2001) 3: 263–286

[6] Polly Wan PO Man, Man Hon Wang, "Efficient and Robust Feature Extraction and Pattern Matching of Time Series by a Lattice Structure"

[7] Chung, F.L., Fu, T.C., Luk, R., Ng, V., 2001. "Flexible time series pattern matching based on perceptually important points. "In: International Joint Conference on Artificial Intelligence Workshop on Learning from Temporal and Spatial Data, pp. 1–7.)

[8] A. Bondu, M. Boullé, B. Grossin, "SAXO : An Optimized Data-driven Symbolic Representation of Time Series" Neural Networks (IJCNN), The 2013 International Joint Conference on.

[9] Gautam Das, King-Ip Lin, Heikki Mannila, "Rule discovery from time series", Autozone Inc., 123 So. Front St., Memphis, TN 38103 USA.

[10] Jessica Lin · Eamonn Keogh · Li Wei, Stefano Lonardi, "Experiencing SAX: a novel symbolic representation of time series. "Data Min Knowl Disc (2007) 15:107–144

[11] Matthew Butler and Dimitar Kazakov, "SAX Discretization Does Not Guarantee Equiprobable Symbols"

AUTHORS' BACKGROUND

Your Name	Title	Research Field	Personal website
Ruidong Xue	master student	Bid data, machine learning	https://gite360.github.io/RuidongXue/
Tingting Zhang	full professor	Sensible Things that Communicate	http://www.miun.se/personal/tingtingzhang
Dehua Chen	associate professor	Database and data warehouse, big data analysis	http://cst.dhu.edu.cn/5d/cb/c3103a24011/page.htm
Jiajin Le	full professor	Data scientific management and software engineering theory and Practice	http://cst.dhu.edu.cn/62/23/c3132a25123/page.htm
Mehrzad Lavassani	Phd candidate	Big data, machine learning	http://www.miun.se/personal/mehrzadlavassani