OEOP: A Novel Algorithm for Periodic Pattern Mining

Jieh-Shan Yeh[1], Szu-Chen Lin[1] and Shueh-Cheng Hu[2]

*[1]Department of Computer Science and Information Management,
Providence University, Taiwan
{jsyeh, g9571043}@pu.edu.tw*

*[2]Department of Computer Science and Communication Engineering,
Providence University, Taiwan
schu @pu.edu.tw*

Abstract

Research on periodic pattern mining has gained a great attention in the past decade. Periodic pattern mining discovers valid periodic patterns in a time-related dataset. This study proposed an efficient 2-D linked structure and the OEOP (One Event One Pattern) algorithm to discover all kinds of valid segments in each single event sequence. Then, this study combines these valid segments found by OEOP into 1-patterns with multiple events, and multiple patterns with multiple events periodic patterns. The experimental results show that the proposed algorithm has good performance and scalability.

Keywords: *Periodic pattern, asynchronous sequence, data mining, pattern mining, sequential pattern.*

1. Introduction

Periodic patterns commonly appear in all kinds of time-series databases. For instance, trajectories of objects, weather, tides, stock market prices, DNA sequences, etc. The discovery of patterns with periodicity is of great importance and has rapidly developed in recent years. The periodic pattern mining models include full-cycle periodic pattern mining [1], segment-wise periodic pattern mining [2], partial periodic pattern mining [3], frequent partial periodic pattern mining [3], and asynchronous periodic pattern mining [4, 5, 6, 7, 8].

This study proposed an efficient linked list structure and the OEOP (One Event One Pattern) algorithm to discover all kinds of valid segments in each single event sequence. Afterwards, by calculating the offsets of the valid 1-pattern segments, the proposed MEOP (Multiple Events One Pattern) algorithm and MEMP (Multiple Events Multiple Patterns) algorithm merged them into multiple-event patterns.

2. Notations and Definitions

Let $E = \{e_1, e_2, \ldots, e_n\}$ be a set of all events. An eventset X is a nonempty subset of E. An eventset with k events is called a k-eventset. A sequence D is an ordered list of eventsets. For example, $E = \{a,b,c,d\}$, $X = \{a,b,c\}$ is a 3-eventset, $D = (\{a,b,c\}\{b,c\}\{a,c,d\}b\{a,c\}d\{a,b,c,d\}a\{a,c,d\}\{a,c\}d\{a,b,c,d\})$ is a sequence.

Definition 1. A ***pattern*** with period l is a nonempty sequence $P = (p_1, p_2, \ldots, p_l)$, where p_1 is an eventset and p_i is either an eventset or *, for $2 \le i \le l$. The symbol * indicates a “don’t care” position. A pattern P is called an ***i-pattern*** if exactly i positions in P contain eventsets.

Definition 2. For two patterns $P=(p_1,p_2,\ldots,p_l)$ and $P^{'}=(p_1^{'},p_2^{'},\ldots,p_l^{'})$ with the same period *l*, $P^{'}$ *is a* **specialization** *of* P (*i.e.*, P is a **generalization** of $P^{'}$) if and only if $p_i \subseteq p_i^{'}$ or $p_i = *$, for $1 \le i \le l$.

Definition 3. For pattern $P=(p_1,p_2,\ldots,p_l)$ with period *l* and a sequence of eventsets $D=(d_1,d_2,\ldots,d_l)$, we say that P ***matches*** D or D ***supports*** P if and only if $p_i \subseteq d_i$ or $p_i = *$, for $1 \le i \le l$. D is also called a ***match*** (or a ***match block***) of P.

Definition 4. Given a pattern P with period *l*, a original sequence D, and k subsequences $D_1,D_2,\ldots,D_k$ of D, if D_i ($1 \le i \le k$) matches P and the distance of D_i and D_{i+1} ($1 \le i \le k-1$) equals 0, the sequence $D_1D_2\ldots D_k$ is called a ***k-segment*** (or a ***continuous matching block*** with the ***repetition*** *k*) of P.

Problem Definition. Given a sequence of eventsets *D*, a minimum repetition *min_rep*, a periodic pattern *P* indicates that there exists a valid subsequence *S* with respect to *P* in *D* and *S* is a set of non-overlapping valid segments, where each valid segment has at least *min_rep* contiguous matches of *P*. ***Periodic pattern mining*** (**PPM**) discovers all periodic patterns in *D*.

3. Proposed Data Structures and Algorithms

This section presents the mining process and introduces the proposed data structures for storing pattern information. First, the **OEOP** algorithm generates all valid 1-pattern segments for each event of the dataset. For different events, by computing the offsets of the valid 1-pattern segments, either the **MEOP** algorithm or the **MEMP** algorithm will merge them into multiple events patterns.

3.1. Proposed Process for Periodic Pattern Mining

The proposed mining process for periodic patterns consists of the 6 steps. Figure 1 illustrates the steps of the proposed mining process for periodic pattern mining.

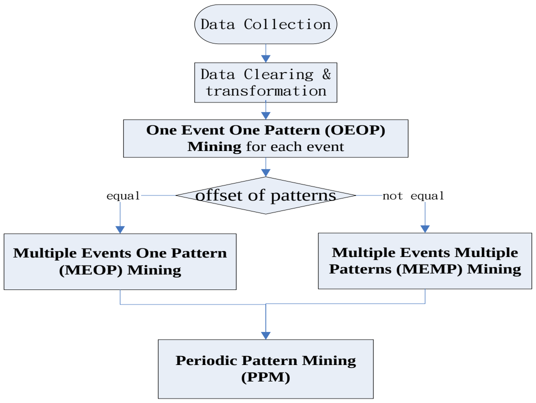
Data Collection
Data Clearing & transformation
One Event One Pattern (OEOP) Mining for each event
offset of patterns
equal
not equal
Multiple Events One Pattern (MEOP) Mining
Multiple Events Multiple Patterns (MEMP) Mining
Periodic Pattern Mining (PPM)

Figure 1. Proposed Process for PPM

3.2. The linked Data Structures

To accelerate the mining process and properly record the pattern information of the list of time instants, we introduce the following three structures, **START** node, **END** node, and **VALID** node.

START node: A structure consists of three fields. The first field, *stime*, saves the starting time instant of a 1-pattern; the second field, *next_s*, is a pointer that links to the next START node; the third field, *list_e*, is a pointer linking to an END node.

END node: A structure consists of four fields. The first field, *etime*, saves an ending time instant of a 1-pattern; the second field, *period*, records the period of the pattern; the third field, *rep_num*, stores the repetition of the pattern; the last field, *next_e*, is a pointer that links to the next END node.

VALID node: A 4-field structure to record a valid 1-pattern. The fields, *stime*, *etime*, *period*, and *rep_num*, indicate the starting time instant, the ending time instant, the period, and the repetition of the 1-pattern, respectively.

3.3. OEOP (One Event One Pattern Mining) Algorithm

Given a sequence of eventsets *D*, for each event *e*, we first generate the list of time instants of *e*, denoted as TL_e. The preliminary goal of OEOP is to discover all valid 1-pattens in TL_e. For each list of time instants TL_e of event *e*, with the minimal repetition *min_rep*, and the maximal period *Lmax*, the OEOP algorithm utilizes the new linked list structures and generates all valid 1-pattern segments of event *e*. The details of OEOP are as follows:

```
OEOP Algorithm
Input: the list of time instants $TL_e$ for event e, min_rep, Lmax
Output: valid segments VS of event e
Method:
L= null ;
       //L : the list of Start node allocate a valid array VS
for each time instant t in $TL_e$ do
{
      allocate a START node X;
     X.stime = t ;
     X.next = null;
     X. list_e = null ;
     L.insert(X) ;        // insert X at the end of L
     for each $X_i$ node L do
   {
       for each $Y_j$ node in $X_i$. list_e do
       {
          if ( t -$Y_j$.etime = $Y_j$.period )
          $Y_j$.etime = t ;    $Y_j$.rep_num++ ;
          if ( t -$Y_j$.etime > $Y_j$.period )
             {
              if ( $Y_j$.rep_num >= min_rep )
                    move VS($Y_j$) ;
                        //insert $Y_j$ at the end of VS array
                    free ($Y_j$) ;  // delete $Y_j$
             }
          if ( t-Xi.stime <= Lmax )
            {
               allocate END node Y;
             Y.etime = t ;
             Y.period = t -$X_i$.stime ;
          Y.rep_num = 2 ;
           $X_i$.list_e.insert (Y) ;
           // insert Y at the end of $X_j$.list_e
          }
     }
}
return VS;
```

4. Experimental Results

All experiments were performed on an Intel Pentium M processor (1.73GHz) PC with 1.50 GB main memory, running the Microsoft Windows XP operating system. The proposed algorithm was implemented in C language.

4.1 Datasets

GenBank Sequences

By using the Entrez interface from the National Center for Biotechnology Information database (http://www.ncbi.nlm.nih.gov/sites/entrez), we randomly selected two protein genbank sequences with different data sizes.

Stock Price Series

Second, we selected the 2008 Taiwan Stock Exchange Capitalization Weighted Stock Index (TAIEX) by Taiwan Stock Exchange Co., Ltd. (TSEC) and Dow Jones Industrial Average Index ($INDU) by Dow Jones & Company. Due to TSEC regulations, the daily change of TAIEX is limited to between -7% and 7%. Therefore, we transformed both TAIEX and $INDU numerical index series to the symbolic series using the following formula:

Change_rate(*i*-th day) = (*i*-th day's index – (*i*-*1*)-th day's index) / (*i*-*1*)-th day's index
Event(*i*-th day) = A, if Change_rate(*i*-th day)>=3%
Event(*i*-th day) = B, if 3% > Change_rate(*i*-th day)>= 1%
Event(*i*-th day) = C, if 1% > Change_rate(*i*-th day)>= -1%
Event(*i*-th day) = D, if -1% > Change_rate(*i*-th day)>= -3%
Event(*i*-th day) = E, if -3% > Change_rate(*i*-th day)

The basic information of each sequence investigated in the experiments is given in Table 1.

Table 1. Basic Information of sequences

Sequence	Length	Event (count)
AJ131352	1104	a:331, t:363, g:217, c:191
X60729	1615	a:474, t:467, g:367, c:307
2008 TAIEX	248	A:16, B:44, C:111, D: 51, E:26
2008 $INDU	252	A:18, B:41, C:119, D:20, E:20

4.2. Numbers of valid segments and sub-sequences

By applying the **OEOP** algorithm on the X60729 GenBank sequence, the 2008 TAIEX sequence and the 2008 $INDU sequence, we obtained valid 1-pattens. Then, by utilizing **MEMP** and **APP** algorithms, we generated valid sub-sequences. Table 2 (a)-(c) list the numbers of valid segments for the X60729 GenBank sequence, the 2008 TAIEX sequence and the 2008 $INDU sequence with *min_rep*=3, *period*=3.

Table 2. Numbers of valid segments of X60729 GenBank, 2008 TAIEX, 2008 $INDU

X60729	number of valid segments
(a, *, *)	65
(t, *, *)	62
(g, *, *)	36
(c, *, *)	18
(a, g, *)	6
(a, *, g)	3
(t, g, *)	4
(c, t, *)	6

(a)

2008 TAIEX	number of valid segments
(B, *, *)	2
(C, *, *)	71
(D, *, *)	7
(C, *, D)	3

(b)

2008 $INDU	number of valid segments
(B, *, *)	2
(C, *, *)	71
(D, *, *)	7
(C, *, D)	3

(c)

4.3. OEOP Results

Figure 2 shows the relationship between length of sequence and running time. Apparently, running time significantly relates to the lengths of sequences. In Figure 3, as expected, the increase in the size of min_rep is observed with decreasing running time, for both X60729 GenBank and 2008 TAIEX sequences.

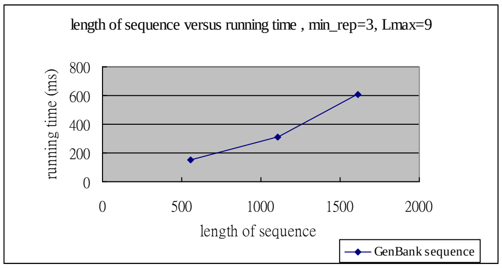

Figure 2. Length of Sequence vs Running Time

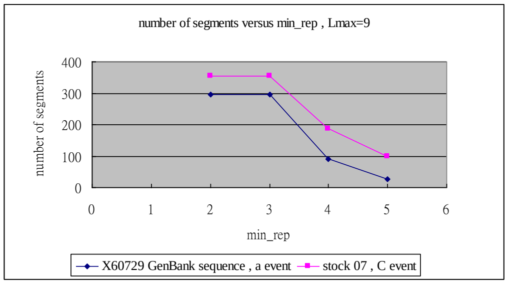

Figure 3. min_rep vs Number of Valid Segments

5. Conclusions

This study proposed an efficient linked list structure and **OEOP** algorithm to discover all kinds of valid segments in each single event sequence. The proposed **MEOP** and **MEMP** algorithms merge 1-patterns into multi-event 1-patterns or multi-event multi-patterns. Implementing these algorithms on real datasets, the experimental results show that these algorithms have good performance and scalability.

Acknowledgements

This work was supported by the National Science Council, Taiwan under grant NSC100-2221-E-126-016 and NSC99-2632-E-126-001-MY3.

References

[1] H. J. Loether and D. G. McTavish, "Descriptive and inferential statistics: an Introduction", Allyn and Bacon **(1993)**

[2] J. Han, W. Gong, and Y. Yin, "Mining segment-wise periodic patterns in time-related databases", In Proceedings of the 4th ACM SIGKDD International Conference Knowledge Discovery and Data Mining **(1998)**, pp. 214-218.

[3] J. Han, G. Dong, and Y. Yin, "Efficient mining partial periodic patterns in time series database", In Proceedings of the 15th International Conference Data Engineering **(1999)**, pp. 106-115.

[4] J. Yang, W. Wang, and P. S. Yu, "Mining asynchronous periodic patterns in time series data", In Proceedings of the 6th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining **(2000)**, pp. 275-279.

[5] J. Yang, W. Wang and P. S. Yu, "Infominer: mining surprising periodic patterns", In Proceedings of ACM SIGKDD International Conference Knowledge Discovery and Data Mining **(2001)**, San Francisco CA, USA, pp. 395-400.

[6] J. Yang, W. Wang and P. S. Yu, "InfoMiner+: mining surprising periodic patterns with gap penalties", In Proceedings of the 2002 IEEE International Conference on Data Mining **(2002)**, pp. 725-728.

[7] J. Yang, W. Wang and P.S. Yu, "Mining asynchronous periodic patterns in time series data", IEEE Transactions on Knowledge and Data Engineering **(2003)**, vol. 15, no. 3, pp. 613-628.

[8] K. Y. Huang and C. H. Chang, "SMCA: a general model for mining asynchronous periodic patterns in temporal databases", IEEE Transactions on Knowledge and Data Engineering **(2005)**, vol. 17, pp. 774-785.

Authors

Jieh-Shan Yeh

Jieh-Shan Yeh obtained his Ph.D degree in Mathematics from the Ohio State University, USA. Dr. Yeh is an associate professor in the Department of Computer Science and Information Management, Providence University, Taiwan, since Sep. 2003. His research interests include, but not limited to, data mining, web mining, cloud computing, information security, XML, and database systems.

Szu-Chen Lin

Szu-Chen Lin received the master's degrees in the Department of Computer Science and Information Management, Providence University, Taiwan in 2008. She is currently employed by Chunghwa Post Co., Ltd., Taiwan.

Shueh-Cheng Hu

Shueh-Cheng Hu is an assistant professor in the Department of Computer Science and Communication Engineering at Providence University, Taiwan. Dr. Hu has been pursuing research in the areas of Web technology, service-based software, e-learning, and e-commerce since 2004. He received his Ph.D. degree in Computer Science from Texas A&M University in 2000; both M.S. and B.A. degrees in Computer Engineering from National Chiao-Tung University, Taiwan, in 1989 and 1987, respectively.