

# SEQUENTIAL TEMPORAL PATTERN MINING IN TIME-INTERVAL BASED EVENT DATA

Kubra Iram<sup>1</sup>, T.R.Mahesh<sup>2</sup>

<sup>1,2</sup>T.John Institute of Technology  
Kubra.iram@gmail.com

## Abstract

In one of the subfield in the world of data mining, sequential mining is one of the imperative subfield. As of late, applications utilizing time interim based occasion data have pulled in impressive endeavors in finding patterns from occasions that hold on for some span. Since the relationship between two interims is inherently unpredictable, how to viably and effectively mine interim based groupings is a testing issue. In this paper, two unusual representations, endpoint representation and endtime representation, are projected to disentangle the preparing of complex connections among occasion interims. In view of the projected representations, three sorts of interim based patterns: fleeting pattern, event probabilistic transient pattern and term probabilistic worldly pattern, are characterized. Likewise, we created two novel calculations, two algorithms mentioned below, to find three sorts of interim based successive patterns. We additionally show three pruning methods to facilitate diminish the pursuit space of the mining process. Exploratory studies demonstrate that both calculations can discover three sorts of patterns proficiently. Besides, we apply proposed calculations to genuine datasets to show the viability and approve the practicality of proposed patterns.

**Keywords:** Sequential Pattern Mining, Endtime, Endpoint Representations, End Time Representation.

\*\*\*

## I. INTRODUCTION

Arrangements of occasions, things, or tokens happening in a requested metric space show up regularly in data and the prerequisite to recognize and dissect continuous subsequences is a typical issue. Consecutive Pattern Mining emerged as a subfield of data mining to concentrate on this field. Consecutive pattern mining is a dynamic exploration theme in data mining in view of its broad pertinence. This kind of utilization dependably considers the request connection and the time issue in our day by day lives. Consecutive pattern mining predominantly manages extricating positive practices that can be utilized to anticipate an occasion in light of the action in the former succession of occasions. Notwithstanding, finding consecutive patterns is a troublesome issue since mining may require producing or examining countless subsequence blends.

In different true situations, a few occasions which inherently tend to persevere for timeframes as opposed to being quick events can't be dealt with as "time focuses." In such cases, the data are normally a grouping of occasions with both begin and complete times. For instance the use of sensor innovation to screen the power utilization of all family unit apparatuses. In particular, power meters are sent to gather machine use log data. The times that every machine is turned on and off can be effortlessly recognized. Clearly, such apparatus use log data are interim based data. Figure 1 beneath demonstrates a specimen time interim data.

SID	event symbol	start time	finish time	interval duration	event sequence
1	A	1	4	3	
	B	3	6	3	
	C	8	12	4	
	D	8	12	4	
2	B	3	6	3	
	C	8	12	4	
	D	8	12	4	
3	C	6	8	2	
	D	6	8	2	
4	A	1	3	2	
	C	6	8	2	
	D	6	8	2	
	E	12	4	2	

Figure 1: Sample Time-Interval Data

## II. EXISTING SYSTEM

Successive pattern mining is a standout amongst the most essential exploration topics in data mining. As of late, various studies on this subject have been directed. These related studies concentrated on time point-based occasion data, and incorporate no understanding of length. Various late studies have explored the mining of interim based occasions. To the best of our insight, the majority of these studies depend on Allen's worldly relations. In any case, Allen's worldly relations are paired in nature, and may show issues to depict the connections among more than two interims. A proper representation and mining strategies for time-interim data is exceptionally urgent.

A few worldly mining calculations were proposed. A few Apriori-kind calculations have been proposed to find fleeting patterns in interim based data. H-DFS changes an occasion arrangement into id-records then consolidations the id-records iteratively to create fleeting patterns. IEMiner utilizes a few streamlining methodologies to lessen the pursuit space and evacuate non-promising applicant transient groupings. Task force was proposed to discover regular worldly patterns in an expansive database. This calculation requires competitor era to decide the relationship for developing patterns from nearby incessant interims. TPrefixSpan involves creating all conceivable hopefuls and after that finding successive occasions and filtering the anticipated databases recursively to find all worldly patterns. HTPM was produced to mine half breed fleeting pattern from occasion successions, which contain both point-form and interim based occasions. Sadasivam adjusted the TPrefixSpan calculation and proposed a clarification strategy to lessen the quantity of database outputs.

Disadvantages of existing system

1. Focused on just point based time occasion data and interim or span of the occasion is not considered.
2. Are ready to recognize connections between just two interims.

### III. PROPOSED SYSTEM

In the determined framework, an alternate representation of the transient time interim data is characterized as appeared in figure 2. The time interim based mining issue is a great deal more exhausting compared point-based prospecting issue. Since two time interims may cover, the relationship among occasion interims is more intricate than that of the occasion focuses. In this anticipate two new representations, endpoint representation and endtime representation, are proposed to viably express fleeting patterns as appeared in figure 2.

SID	event symbol	start time	finish time	event sequence	(endpoint sequence endtime sequence)
1	A	2	7		$(A^+ (B^+ C^+) A^- B^- C^- D^+ E^+ E^- D^-)$ $(2 \ 5 \ 5 \ 7 \ 10 \ 12 \ 16 \ 18 \ 20 \ 22)$
1	B	5	10		
1	C	5	12		
1	D	16	22		
1	E	18	20		
2	B	1	5		$(B^+ B^- D^+ (E^+ F^+) (E^- F^-) D^-)$ $(0 \ 5 \ 8 \ 10 \ 10 \ 13 \ 13 \ 14)$
2	D	8	14		
2	E	10	13		
2	F	10	13		
3	A	6	12		$(A^+ B^+ A^- (B^- D^+) E^+ E^- D^-)$ $(6 \ 7 \ 12 \ 14 \ 14 \ 17 \ 19 \ 20)$
3	B	7	14		
3	D	14	20		
3	E	17	19		
4	B	8	16		$(B^+ B^- A^+ A^- D^+ E^+ E^- D^-)$ $(8 \ 10 \ 13 \ 16 \ 20 \ 21 \ 22 \ 23)$
4	A	18	21		
4	D	24	28		
4	E	25	27		

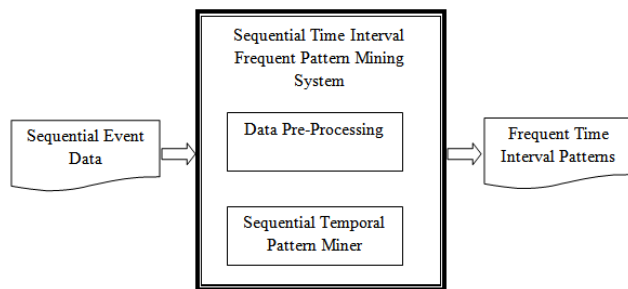
Figure 2: DeterminedendPoint&endTime representation of time interim data

At that point, Allen's 13 transient relations are communicated utilizing the projected representations which are represented in figure 3.

temporal relation	inversed relation	pictorial example (s: starting time, f: finishing time)	endpoint representation	endtime representation
A before B	B after A		$A^+ A^- B^+ B^-$	$(A^+ A^- B^+ B^-)$ $(s_A \ f_A \ s_B \ f_B)$
A overlaps B	B overlapped-by A		$A^+ B^+ A^- B^-$	$(A^+ B^+ A^- B^-)$ $(s_A \ s_B \ f_A \ f_B)$
A contains B	B during A		$A^+ B^+ B^- A^-$	$(A^+ B^+ A^- B^-)$ $(s_A \ s_B \ f_A \ f_B)$
A starts B	B started-by A		$(A^+ B^+) B^- A^-$	$(A^+ B^+) B^- A^-$ $(s_A \ s_B \ f_B \ f_A)$
A finished-by B	B finishes A		$A^+ B^+ (A^- B^-)$	$(A^+ B^+ (A^- B^-))$ $(s_A \ s_B \ f_A \ f_B)$
A meets B	B met-by A		$A^+ (A^- B^+) B^-$	$(A^+ (A^- B^+) B^-)$ $(s_A \ f_A \ s_B \ f_B)$
A equal B	B equal A		$(A^+ B^+) (A^- B^-)$	$((A^+ B^+) (A^- B^-))$ $(s_A \ s_B \ f_A \ f_B)$

Figure 3: Allen's transitive relation

#### IV. SYSTEM IMPLEMENTATION



**Figure 4:** System Architecture of projected Sequential Time Interim Frequent Pattern Mining System

Consecutive Occasions are the arrangement of occasions happening aimlessly time interims. Every occasion has a begin time and the completion time and distinctive occasions execution interims can cover i.e. occasions running in parallel. These occasions are recorded in a discrete time interims as a grouping of occasions. Case of a Successive Occasions: Programming Advancement Life Cycle of various activities Give us a chance to accept that Organization XYZ is an expansive programming administrations organization which gives programming answer for various customers and executes numerous activities. The organization has received Iterative and Incremental SDLC model for the advancement of its tasks. So as to enhance the quality and velocity of execution of tasks, the organization examinations the execution of various cycles in the ventures and stages utilized as a part of these emphases of the activities which were effective before.

**1. Data Parser:** This module parses the information consecutive occasion record and concentrates the occasions with time stamp and makes the rundown of groupings in memory for further handling.

In the data record,

<> image is utilized for determining time interim which are augmentation by single time interim.

\* is utilized determine end of the occasions.

# is utilized determine the end of succession.

Occasions are spoken to by the letter sets like A, B , C and so forth as identifiers.

More than one occasion can happen is same time interim.

On the off chance that the occasion slips by for more than one time interim then the occasion identifier is rehashed in successive time interims till the time interim it closes.

The parser peruses the information document arrangement by succession. For every grouping read, it first concentrates time interim said in the middle of "<" and ">". At that point it understands all the occasion identifiers which are isolated by space character till "\*" image is come to. The occasion identifiers read are put away in the Hash Map data structure

with Time Interim as Key and Occasion Identifier list as qualities. This procedure is rehashed till all the time interims and occasions are put away for a succession.

Essentially all the arrangements in the information document are perused and put away in the Grouping Database which is an in-memory data structure.

**2. End Point Converter:** This module changes over the parsed data to proposed End Point design. The End Point group gives exact Occasion Begin and End time interims rather than rehashed occasion id as present in the data arrangement document. The Occasion Begin is spoken to by "+" and Occasion End is spoken to by "- ". It is a conservative representation of the occasion arrangements.

**3. Applicant Pattern Extractor:** This module removes competitor succession of occasions which fulfill the state of covering and same end point. Just these applicant patterns are considered for finding the real backing of the occasions which will lessen the seeking time.

**4. Pattern Bolster Discoverer:** This module computes support for every hopeful patterns from the info document/database. The backing of the pattern is the include of its event the data document. The competitor patterns are extricated by perusing the groupings in end point position. For any occasion begin, the occasion end point is likewise perused and time interims of begin and end are recorded. This occasion begin and end pattern is looked in different groupings for its event. On the off chance that the pattern happens in another succession then its bolster number is augmentation by 1. Thus the pattern is looked in all the successions and its bolster tally is increased by 1 if the pattern is found.

In the first place the bolster mean every individual occasion is resolved. At that point utilizing same bolster checking strategy pattern having blend of occasions is found. This procedure is rehashed till all blend of occasions present in the successions are sought and their bolster check are figured.

**5. Continuous Pattern Extractor:** This module takes the rundown of applicant patters with their bolster check and the Minimum\_Support esteem said by the client and prunes all the rare patterns and yields just the incessant successive worldly patterns to the client. Every applicant patterns bolster check is contrasted and the utilization determined Minimum\_Support. In the event that the bolster number of the hopeful pattern is not exactly the Minimum\_Support then that pattern is pruned. Rehashing this procedure for all the patterns, just the patterns fulfilling the Minimum\_Support is kept.

### Algorithm to generate End Point Representation of Data

Input: Temporal Sequential Data File  
Output: End Point Representation of Data

1. Read Data File
2. FOR each line in Data File
3.   Read Event
4.   Read TimePoints
5.   EventList.Add(Event, TimePoints)
6. END FOR
- 7.
8. FOR each TimePoint
9.   IF Event starts in TimePoint
10.     Event+ = TimePoint
11.   ELSE IF Event ends in TimePoint
12.     Event- = TimePoint
13. END IF

### Algorithm to find frequent temporal interval patterns

Input: End Point Representation of Data, MinimumSupport  
Output: Frequent Temporal Frequent Interval Patterns

1. FOR each EndPoint sequence in EndPointRepresentation
2.   SortEventsInOccuranceOrder
3.   Node = EndPoint
4.   IF PrefixSpan.Root = NULL THEN
5.     PrefixSpan.Root = Node
6.   ELSE
7.     PrefixSpan.AddNode(Node)
8.   END IF
- 9.
10. FOR each Branch in the PrefixSpanTree
11.   Pattern = EventsInBranch
12.   IF Pattern contains Both Start and End Events AND Pattern.Support >= MinimumSupport
13.     FrequentPattern.Add(Pattern)
14.   END IF
15. END FOR

## V. RESULT ANALYSIS

The screenshot displays the 'Sequential Temporal Interval Pattern Mining' application. It features a 'Load Interval Data' button, a text area for input data, and a 'Generate EndPoint Data' button. Below these are two search buttons: 'Find Interval Based Freq Patterns' and 'Find Point Based Freq Patterns', both with a 'Minimum Support' of 5. The results are shown in two panes. The left pane lists point-based frequent patterns such as 'E+\*E-\* [SUP: 6]' and 'D+\*E+\*E-\*D-\* [SUP: 6]'. The right pane lists interval-based frequent patterns such as 'E+\* [SUP: 6]' and 'E+\*D-\* [SUP: 6]'. The 'EndPointRepresentation' section shows a list of events with their time points, such as 'A+(2) C+(5) B+(5) A-(7) B-(10) C-(12) D+(16) E+(18) E-(20) D-(22)'.

In the above analysis it is seen that point based frequent patterns are more than time based event patterns and also the patterns in point based occurs only on point and does not

take time intervals and overlapping in consideration whereas the time based pattern mining takes interval and also the overlapping in consideration and yield better results.

The screenshot shows the 'Output - TemporalIntervalPatternMining (run)' window. It contains two sections of statistics. The first section, 'POINT BASED PREFIXSPAN - STATISTICS', shows a total time of ~10 ms, 127 frequent sequences, and a maximum memory usage of 8.39990234375127 mb. The second section, 'INTERVAL BASED PREFIXSPAN - STATISTICS', shows a total time of ~2 ms, 11 frequent sequences, and a maximum memory usage of 6.92086791992187511 mb.

Here we can simulate and compare both pattern minings and can see that the total time, frequent sequences and maximum memory is less in time interval based mining when compared to point based, thus we can tell that interval based mining is more efficient and consumes less time and memory than point based mining.

## CONCLUSION

Mining fleeting patterns from time interim based data is a critical subfield in data mining. Be that as it may, the perplexing connections among interims expand the trouble of outlining a proficient calculation. In this projected paper, two novel representations, endpoint representation and endtime representation, are projected to cure this basic issue. In view of these two representations, we propose three sorts of patterns: fleeting pattern, event probabilistic transient pattern, and duration probabilistic worldly pattern, to depict the connection among interims and the likelihood of the happening time and length of every interim. In addition, the calculations of two algorithms mentioned are created to effectively discover temporal pattern and probabilistic fleeting patterns, individually. The additionally propose a few pruning procedures to viably lessen the inquiry space. The trial concentrates on demonstrate that two mentioned algorithms are effective and functional. Both the runtime and memory use of time interval based algorithms beats those of the best in class calculations. What's more, time interval based algorithms can successfully find both the event probabilistic fleeting patterns and the term probabilistic transient patterns. There are a few fascinating issues that could be concentrated further. Mining shut worldly patterns could utilize smaller results to safeguard the same expressive force as transient pattern mining. Incremental fleeting pattern mining is additionally a critical issue which has numerous realworld applications.

## REFERENCES

- [1] R. Agrawal and R. Srikant, "Mining Sequential Patterns," The 11th International Conference on Data Engineering (ICDE'95), pp.3-14, 1995.
- [2] J. Allen, "Maintaining Knowledge about Temporal Intervals," Communications of ACM, vol. 26, issue 11, pp.832-843, 1983.
- [3] F. Chen, J. Dai, B. Wang, S. Sahu, M. Naphade and C. Lu, "Activity Analysis Based on Low Sample Rate Smart Meters," The 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11), pp. 240-248, 2011.
- [4] J. Chen, "An Up Down Directed Acyclic Graph Approach for Sequential Pattern Mining," IEEE Transactions on Knowledge and Data Engineering, vol. 22, no. 7, pp.913-928, 2010.
- [5] Y. Chen, J. Jiang, W. Peng and S. Lee, "An Efficient Algorithm for Mining Time Interval-based Patterns in Large Databases," The 19th ACM International Conference on Information and Knowledge Management (CIKM'10), pp. 49-58, 2010.
- [6] Y. Chen, C. Chen, W. Peng and W. Lee, "Mining Correlation Patterns among Appliances in Smart Home Environment," The 18th Pacific-Asia Conference in Knowledge Discovery and Data Mining, Advances in Knowledge Discovery and Data Mining, (PAKDD'14), pp. 210-221, 2014.
- [7] F. Hoppner, "Finding informative rules in interval sequences," Intelligent Data Analysis, vol. 6, no. 3, pp. 237-255, 2002.
- [8] P. Kam and W. Fu, "Discovering Temporal Patterns for Interval- based Events," International Conference on Data Warehousing and Knowledge Discovery (DaWaK'00), vol. 1874, pp. 317-326, 2000.
- [9] H. Kim, M. Marwah, M. Arlitt, G. Lyon and J. Han, "Unsupervised disaggregation of low frequency power measurements," The 11th SIAM International Conference on Data Mining (SDM'11), pp. 747-758, 2011.
- [10] J. Kolter, and M. Johnson, "REDD: A public data set for energy disaggregation research," KDD workshop on Data Mining Applications in Sustainability (SustKDD'11), pp. 1-6, 2011.
- [11] S. Laxman, P. Sastry and K. Unnikrishnan, "Discovering Frequent Generalized Episodes When Events Persist for Different Durations," IEEE Transactions on Knowledge and Data Engineering, vol. 19, no. 9, pp. 1188-1201, 2007.
- [12] M. Leleu, C. Rigotti, J. Boulicaut and G. Euvarad, "Constraint- Based Mining of Sequential Patterns over Datasets with Consecutive Repetitions," The 7th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD'03), pp. 303-314, 2003.
- [13] Y. Li, J. Bailey, L. Kulik and J. Pei, "Mining Probabilistic Frequent Spatio-Temporal Sequential Patterns with Gap Constraints from Uncertain Databases," The 13th International Conference on Data Mining (ICDM'13), pp. 448-457, 2013.
- [14] M. Lin and S. Lee, "Fast Discovery of Sequential Patterns by Memory Indexing and Database Partitioning," Journal of Information Sciences and Engineering, vol. 21, no. 1, pp. 109-128, 2005.
- [15] B. Liu, Y. Yang, G. Webb and J. Boughton, "A Comparative Study of Bandwidth Choice in Kernel Density Estimation for Naive Bayesian Classification," The 13th Pacific-Asia Conference in Knowledge Discovery and Data Mining, Advances in Knowledge
- [16] F. Masegla, F. Cathala and P. Poncelet, "The PSP Approach for Mining Sequential Patterns," European Conference on Principles of Data Mining and Knowledge Discovery (PKDD'01), vol.1510, pp176-184, 1998.
- [17] F. Morchen and D. Fradkin, "Robust mining of time intervals with semi-interval partial order patterns," Proceedings of the SIAM International Conference on Data Mining (SDM'10), pp.315-326, 2010.
- [18] F. Morchen and A. Ultsch, "Efficient Mining of Understandable Patterns from Multivariate Interval Time Series," Data Mining Knowledge Discovery, vol. 15, no. 2, pp.181-215, 2007.
- [19] M. Muzammal and R. Raman, "Mining Sequential Patterns from Probabilistic Databases," The 15th Pacific-Asia Conference in Knowledge Discovery and Data Mining, Advances in Knowledge Discovery and Data Mining, (PAKDD'11), pp. 210-221, 2011.
- [20] P. Papapetrou, G. Kollios, S. Sclaroff and D. Gunopulos, "Discovering frequent arrangements of temporal intervals," International Conference on Data Mining (ICDM'05), pp. 354-361, 2005.