DOI: 10.15849/IJASCA.221128.07

Int. J. Advance Soft Compu. Appl, Vol. 14, No. 3, November 2022
Print ISSN: 2710-1274, Online ISSN: 2074-8523

Framework to Mine XML Format Event Logs

Ang Jin Sheng, Jastini Mohd Jamil, and Izwan Nizal Mohd Shaharanee

School of Quantitative Sciences,
Universiti Utara Malaysia, 0601 Sintok, Kedah, Malaysia
e-mail:
angjinsheng@gmail.com
jastini@staf.uum.edu.my
nizal@staf.uum.edu.my

Abstract

A lot of applications including event logs and web pages uses XML format for utilizing, keeping, transferring and displaying data. Thus, volume of data expressed in XML has increase rapidly. Numerous research has been done to extract and mine information from XML documents. Mining XML documents allows an understanding to the architecture and composition of XML documents. Generally, frequent subtree mining is one of the methods to mine XML documents. Frequent subtree mining searches the relation between data in a tree structured database. Due to the architecture and the composition of XML format, normal data mining and statistical analysis difficult to be performed. This paper suggests a framework that flattens and converts tree structured data into structured data, while maintaining the information of architecture and the composition of XML format. To gain more information from event logs, converting into structured data from semi-structured format grants more ability to perform variety data mining techniques and statistical test.

Keywords: *Flatten Sequential Structure Model, XML Format Event Logs, Data Mining, Statistical Analysis.*

1 Introduction

The size of eXtensible Markup Language (XML) is growing exponentially daily with the new technology and capabilities in keeping and searching those data. Moreover, one of the formats used for data representation and transaction in the World Wide Web is XML format [4]. Thus, they hold an enormous percentage (58%) in the web [18]. Besides, event happens in business process is captured in XML format as well.

Business process is a collection of cases that has occurred in a business structurally with a target to generate a desired outcome[1]. Business process management system (BPMS) has developed to aid these business processes. During the business process happens, the event logs are produced by the BPMS. XML format is the most popular format to store and capture event logs [22; 26]. These event logs can be extracted for mining or analysing purpose to gain more insights from the business processes [41]. For instance, Halal industry utilizes XML format event logs to ensure the quality of Halal products during the business processes happen [5]. However, the data mining and

Received 15 August 2022; Accepted 15 October 2022

analysing process becomes more difficult due to the complicated of data architecture and composition of XML format [29].

To gain knowledge from the XML format data, dozens of researches on frequent subtree mining (FSM) has been done in these past few years [7; 23; 46]. The purpose of FSM is to search for interesting relation between transactions in the tree database. Rules about relation between transactions are generated by FSM based on minimum support fixed by user. Nevertheless, the performance of FSM will drop when the rules are produced neither interesting nor useful. Shaharanee and Jamil [32] propose that removing variables that are not relevant can reduce the rules that are not interesting. Moreover, the XML documents' structure properties usually neglected by FSM. In a nutshell, framework that can apply statistical analysis and mine XML format event logs without ignore structural properties of XML format is essential to obtain more information from the event logs.

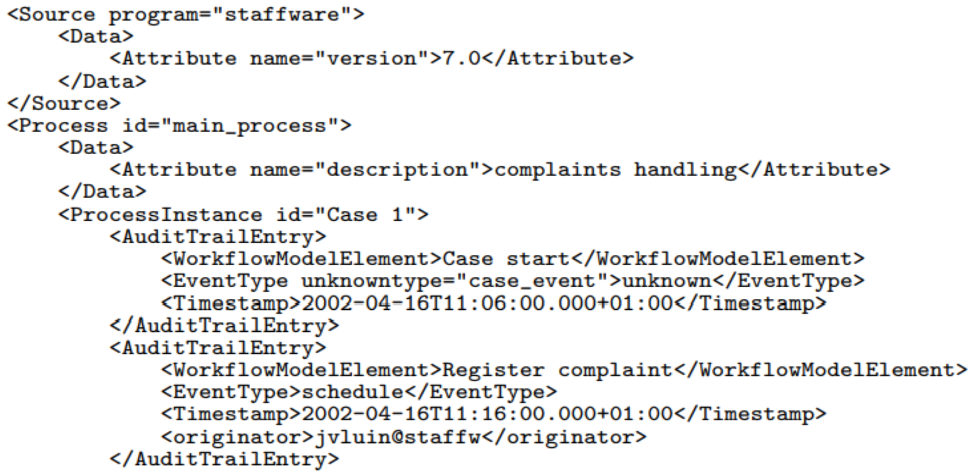
2 Literature Review

2.1. Business Process

The definition of business process is the techniques to describe the approach to fulfill specific action in an organization [14]. Researchers [16; 24; 40] believe that business process is an entire series of cases and actions to reach business goals. Therefore, business process plays an essential role to comprehend how an organization operates [44]. Operational business process such as customer relationship management, account management and invoice management are executed and taken care by business process management system (BPMS). In other words, BPMS is invented to ensure routine business process can happen smoothly.

2.2. Event Logs

Event logs are log files that captured a series of events generated by BPMS [35]. Most of the XML format event logs appear in two standards, Macromedia eXtensible Markup Language (MXML) and eXtensible Event Stream (XES). MXML standard starts in year 2003 and it is the earliest standard for event logs in XML format [39]. Fig 1 illustrates the examples of event of MXML standard whereas Fig 2. Shows the meta model of MXML standard [42]. Then, XES standard succeed MXML standard during the year 2009. XES standard is widely use and famous recently due to it became IEEE standard in the year of 2016 [20]. The example of event log in XES standard is illustrated in Fig 3 and meta model structure of XES standard is shown in Fig 4.



```
<Source program="staffware">
    <Data>
        <Attribute name="version">7.0</Attribute>
    </Data>
</Source>
<Process id="main_process">
    <Data>
        <Attribute name="description">complaints handling</Attribute>
    </Data>
    <ProcessInstance id="Case 1">
        <AuditTrailEntry>
            <WorkflowModelElement>Case start</WorkflowModelElement>
            <EventType unknowntype="case_event">unknown</EventType>
            <Timestamp>2002-04-16T11:06:00.000+01:00</Timestamp>
        </AuditTrailEntry>
        <AuditTrailEntry>
            <WorkflowModelElement>Register complaint</WorkflowModelElement>
            <EventType>schedule</EventType>
            <Timestamp>2002-04-16T11:16:00.000+01:00</Timestamp>
            <originator>jvluin@staffw</originator>
        </AuditTrailEntry>
```

Fig 1: Snapshot of MXML Standard Event Log

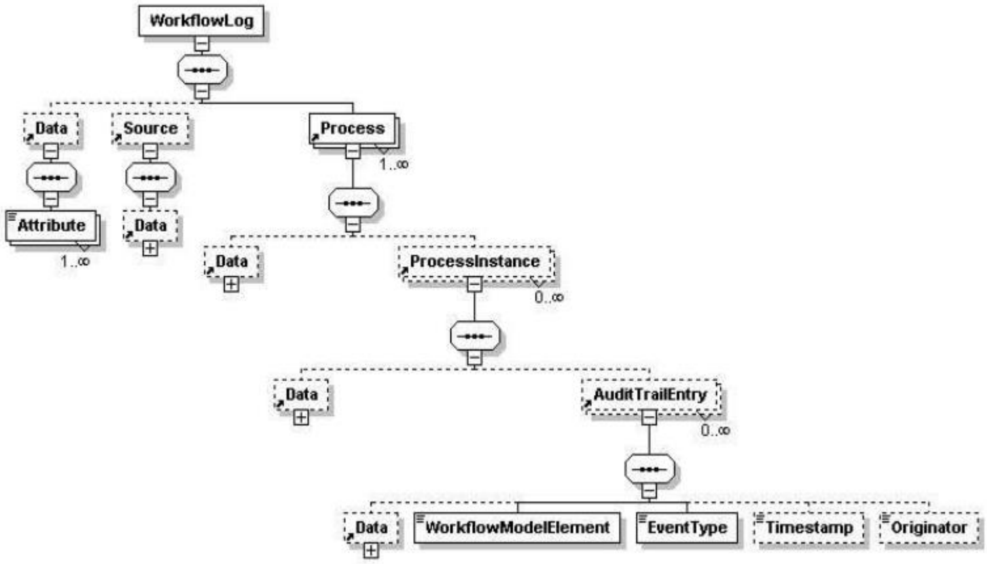
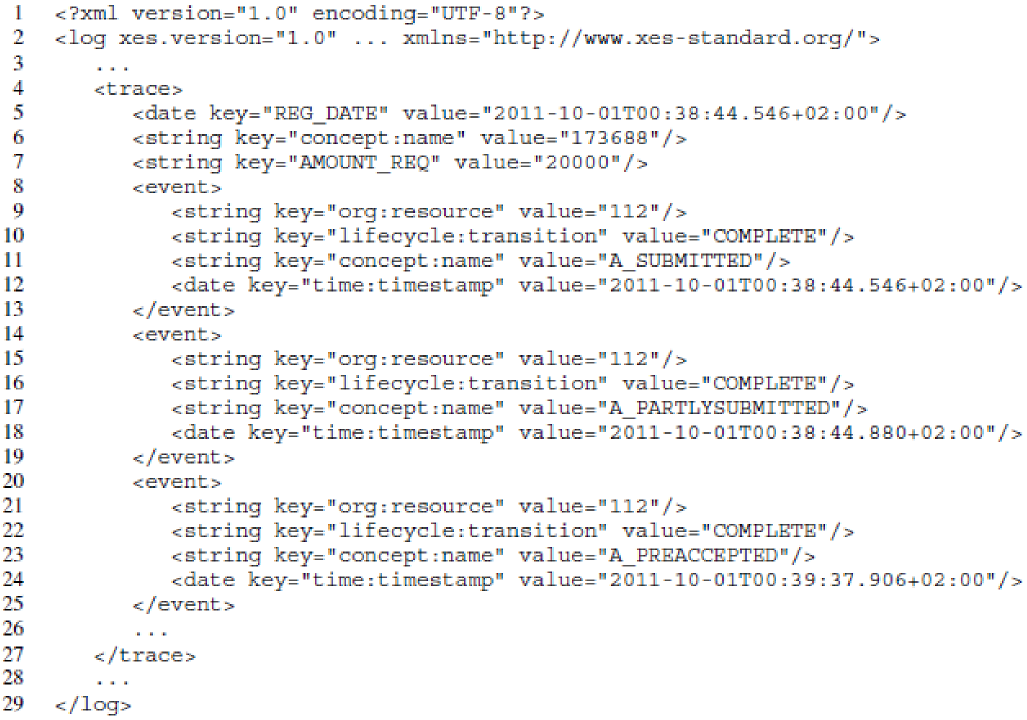


Fig 2: Meta Model of MXML standard



```
<?xml version="1.0" encoding="UTF-8"?>
<log xes.version="1.0" ... xmlns="http://www.xes-standard.org/">
   ...
   <trace>
      <date key="REG_DATE" value="2011-10-01T00:38:44.546+02:00"/>
      <string key="concept:name" value="173688"/>
      <string key="AMOUNT_REQ" value="20000"/>
      <event>
         <string key="org:resource" value="112"/>
         <string key="lifecycle:transition" value="COMPLETE"/>
         <string key="concept:name" value="A_SUBMITTED"/>
         <date key="time:timestamp" value="2011-10-01T00:38:44.546+02:00"/>
      </event>
      <event>
         <string key="org:resource" value="112"/>
         <string key="lifecycle:transition" value="COMPLETE"/>
         <string key="concept:name" value="A_PARTLYSUBMITTED"/>
         <date key="time:timestamp" value="2011-10-01T00:38:44.880+02:00"/>
      </event>
      <event>
         <string key="org:resource" value="112"/>
         <string key="lifecycle:transition" value="COMPLETE"/>
         <string key="concept:name" value="A_PREACCEPTED"/>
         <date key="time:timestamp" value="2011-10-01T00:39:37.906+02:00"/>
      </event>
      ...
   </trace>
   ...
</log>
```

Fig 3: Example of XES Standard Event Log

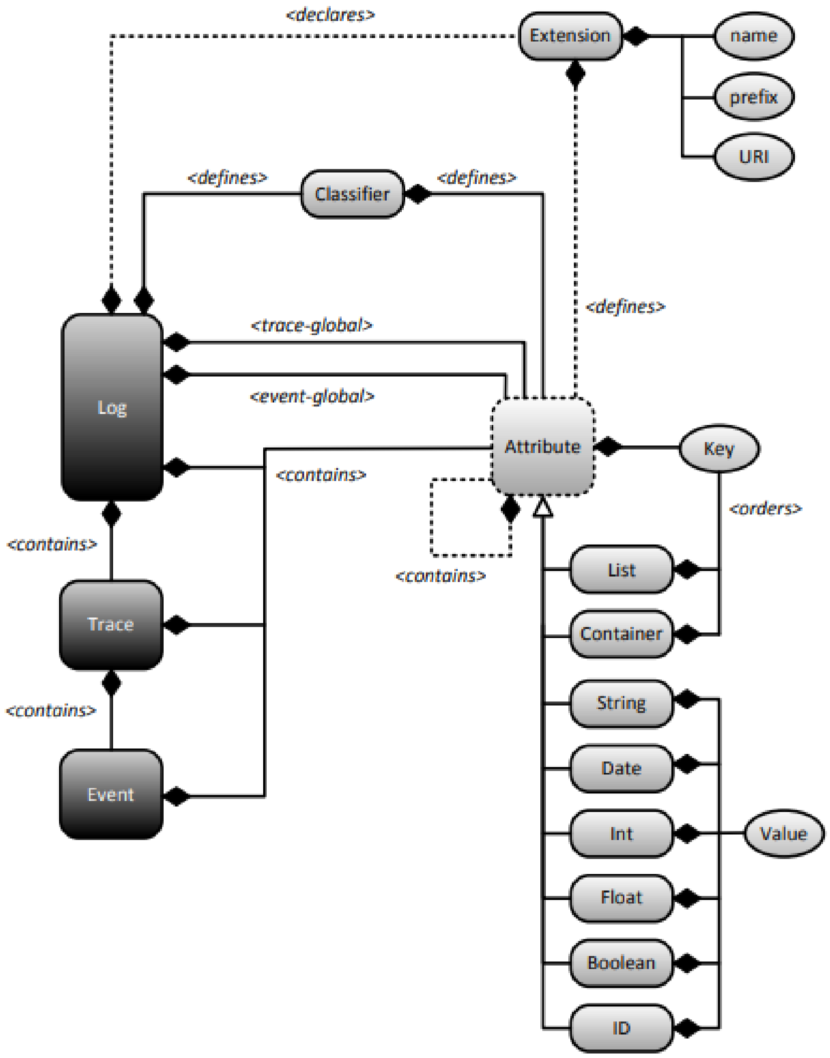


Fig 4: Meta Model of XES Standard

2.3. Frequent Subtree Mining (FSM)

A tree includes root, vertex or node, vertex label, edge and edge label. The relationship become parent and child in between the vertices form when a direct edge in connected between two vertices. XML format log can be modelled as ordered labelled and rooted trees [47] as it has the characteristics of ordered, labelled and rooted trees such as child order is essential and every node contains label. Therefore, XML format event log can be mined through frequent subtree mining (FSM). FSM is a method that can discover an equal or greater number of times of a subtree occur in a tree database [31]. Different algorithms of FSM developed by past researchers are outlined in Table 1. There are 4 different types of structures in the tree which consist of tree that organized orderly, unorderly, tree that combine both ordered and unordered structure and lastly the free tree. For the tree that organized orderly, every node's arrangement and places are essential in tree structure mining. Numerous frequent subtree mining algorithm has been invented to mine induced, closed, embedded, maximal and phylogenetic subtrees.

Table 1: FSM Algorithms

Type of tree mining	Algorithm	Authors	Maximal	Closed	Induced	Embedded	Phylogenetic
Free Tree Mining	FreeTreeMiner	[8]			∨		
	FreeTreeMiner	[30]			∨		
	HybridTreeMiner	[9]			∨		
	GASTON	[28]			∨		
	Phylominer	[48]					∨
	EvoMiner	[15]					∨
Unordered Tree Mining	TreeFinder	[38]	∨			∨	
	uFreqT	[27]			∨		
	PathJoin	[45]	∨		∨		
	Unot	[3]				∨	
	CousinPair	[33]				∨	
	RootedTreeMiner	[10]			∨		
	SLEUTH	[46]				∨	
	Uni3	[17]				∨	
	BEST	[12]				∨	
	IRTM	[25]				∨	
	BOSTER	[13]			∨		
Ordered Tree Mining	FREQT	[2]			∨		
	Chopper and Xspanner	[43]				∨	
	AMIOT	[19]			∨		
	TreeMiner	[47]				∨	
	MB3-Miner	[6]				∨	
	IMB3-Miner	[36]			∨	∨	
Hybrid Tree Mining	CMTreeMiner	[11]	∨	∨	∨		
	TRIPS and TIDES	[37]			∨	∨	
	POTMINER	[21]			∨	∨	

3 Proposed Framework

A new framework to mine XML format event logs is proposed in this study. Fig 5 shows the proposed framework. The proposed framework able to flatten tree-structured data, then converts into structured data. Thus, a series of data mining and statistical analysis can be performed on the event logs. The motivation to develop this framework is to find out ways to combine data mining and statistical measurement to generate more useful and interesting rules compare to FSM or association rule mining. Interesting rules can be understood as the rules that are not excessive or repetitious and have statistical support. Hypothesis development, sampling process, model building

Received 15 August 2022; Accepted 15 October 2022

and statistical analysis techniques are required to verify the quality of rules generated by FSM or association rule. Thus, misleading and excessive rules can be removed to ensure the quality of decision made by decision maker. The detail phases that happen in the framework are elaborated as following.

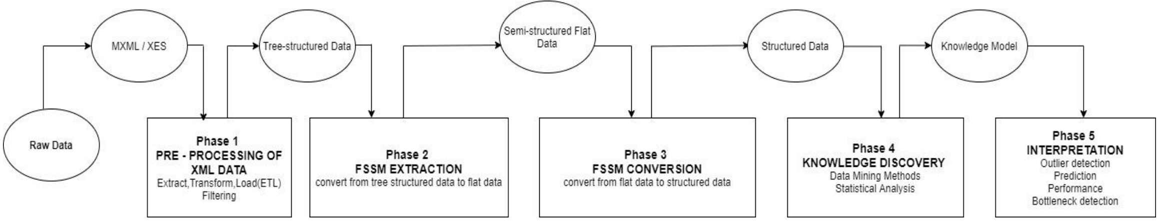


Fig 5: Proposed Framework

Phase 1: XML Data Pre-processing

Before conducting any data mining or statistical analysis, raw data in actual world need to be pre-processed. The major cause pre-processed the data is because the data in actual world usually is not in consistent, complete and arranged manner [34]. Before loading data to the destination, data is transformed through Extract, transform and load (ETL) process. ETL plays a substantial role to ensure the event logs in XES standard XML format as the event logs may appear in different formats. The main reason of transformation event log format into XES standard is XES is the IEEE standards for event log and extensively used in most recently. Filtering is the process where the unwanted data being get rid of. Therefore, corrupted transactions or data that are not related to transactions or business processes will be removed in this phase. After the process of filtering and ETL, pre-processed data is prepared for the next phase.

Phase 2: Data Extraction (Flatten Sequential Structure Model (FSSM))

The next following 2 phases are phases of Flatten Sequential Structure Model (FSSM), extraction phase and conversion phase. The FSSM extraction phase is to flatten the tree structured data without ignoring the structural position of the data. An illustration of tree structure database with two set of datasets is displayed in Fig 6. 2 transactions labelled as t_1 and t_2. Different structures of subtree are used to illustrates how transactions or data is flatten from tree structure through FSSM extraction phase. Architecture information of each node in the tree-structured database are conserved and captured through FSSM extraction phase. Table 2 shows the example of data after gone through FSSM extraction process. A top down then left right sequence arranged and viewed in tree structured data. A backtrack to previous node is required when '-1' shows in Table 2. Going back to parent node is required before advance to right side of the node.

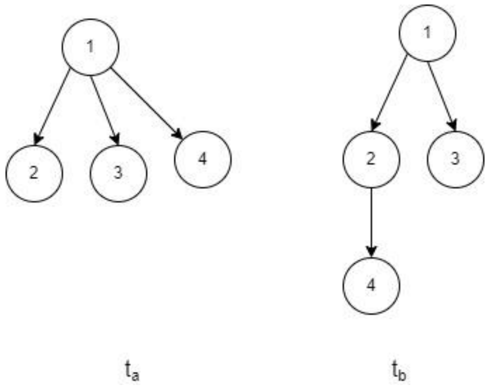


Fig 6: Tree Structure Database

Table 2: Example of FSSM Extraction Flat Data

T_e	**x_0**	**x_1**	**x_2**	**x_3**	**x_4**	**x_5**	**x_6**
t_a	1	2	-1	3	-1	4	-1
t_b	1	2	3	-1	-1	4	0

Phase 3: Data Conversion (Flatten Sequential Structure Model (FSSM))

After extraction phase, conversion phase is the next phase. The conversion phase transforms the flatten data into a structured table. Thus, more application of data mining techniques and statistical test are enabled. Table 3 illustrates the instance of data format after gone through FSSM conversion process. The flatten data is categorized according to the unique variables to become structured data.

Table 3: Example of FSSM Conversion Structured Data

	1	**2**	**3**	**4**
t_a	t_a1	t_{a2}	t_a3	t_a4
t_b	t_b1	t_b2	t_b3	t_b4

Phase 4: Knowledge discovery

After two phases of FSSM, various data mining and statistical analysis methods can be performed in this phase. For instance, the dependent variable can be categorized corresponding to different business requirements and be used for classification purposes. ANOVA test or t-test can be performed to understand the difference of performance between 2 or more groups. For example, machines that produce same products can be measured through ANOVA or t-test to ensure the consistency of the machine. Relationship between variables can be determined through Pearson test or correlation test. To ensure the interestingness or usefulness of rules generated by FSM or association rule, statistical analysis can be conducted to filter variables that are unimportant.

Phase 5: Interpretation

The outcome produced by prior phase will be explained in comprehensible way by the domain experts in this phase. Then, decision maker can make a good decision accordingly.

4 Conclusion

In short, a framework is recommended in this research articel that it allows a range of statistical test and data mining techniques applied in event logs. In order to apply these techniques, the proposed framework flattens and converts the XML format event log data into a structured data from a tree structured format. Therefore, direct application of data mining methods and statistical test in event logs are enabled. By applying this framework into mining event logs, decision makers understand more about business processes through event logs. Therefore, a higher quality judgement can be made through mining these event logs. This will bring more profit to business and reduce loss in business. This framework can be tested to mine event logs in the future practically through different type of dataset including simulated and real dataset.

ACKNOWLEDGEMENTS

The authors wish to acknowledge the financial support provided by the Fundamental Research Grant Scheme with (FRGS/1/2018/ICT02/UUM/02/5), Universiti Utara Malaysia, Sintok, Kedah, Malaysia, for conducting this research.

References

[1] Aguilar-Saven, R. S. (2004). Business process modelling: Review and framework. *International Journal of production economics, 90*(2), 129-149.

[2] Asai, T., Abe, K., Kawasoe, S., Sakamoto, H., Arimura, H., & Arikawa, S. (2004). Efficient substructure discovery from large semi-structured data. *IEICE TRANSACTIONS on Information and Systems, 87*(12), 2754-2763.

[3] Asai, T., Arimura, H., Uno, T., & Nakano, S.-I. (2003). *Discovering frequent substructures in large unordered trees.* Paper presented at the International Conference on Discovery Science.

[4] Bača, R., Krátký, M., Holubová, I., Nečaský, M., Skopal, T., Svoboda, M., & Sakr, S. (2017). Structural XML query processing. *ACM Computing Surveys (CSUR), 50*(5), 1-41.

[5] Belkhatir, M., Bala, S., & Belkhatir, N. (2020). Business process re-engineering in supply chains examining the case of the expanding Halal industry. *arXiv preprint arXiv:2004.09796.*

[6] Chang, E., Tan, H., Dillon, T. S., Hadzic, F., & Feng, L. (2005). *MB3-Miner: Efficient mining eMBedded subTREEs using tree model guided candidate generation.* Paper presented at the Proceedings of the First International Workshop on Mining Complex Data (MCD).

[7] Chi, Y., Muntz, R. R., Nijssen, S., & Kok, J. N. (2005). Frequent subtree mining–an overview. *Fundamenta Informaticae, 66*(1-2), 161-198.

[8] Chi, Y., Yang, Y., & Muntz, R. R. (2003). *Indexing and mining free trees.* Paper presented at the Third IEEE International Conference on Data Mining.

[9] Chi, Y., Yang, Y., & Muntz, R. R. (2004). *HybridTreeMiner: An efficient algorithm for mining frequent rooted trees and free trees using canonical forms.* Paper presented at the Proceedings. 16th International Conference on Scientific and Statistical Database Management, 2004.

[10] Chi, Y., Yang, Y., & Muntz, R. R. (2005). Canonical forms for labelled trees and their applications in frequent subtree mining. *Knowledge and Information Systems, 8*(2), 203-234.

[11] Chi, Y., Yang, Y., Xia, Y., & Muntz, R. R. (2004). *Cmtreeminer: Mining both closed and maximal frequent subtrees.* Paper presented at the Pacific-Asia Conference on Knowledge Discovery and Data Mining.

[12] Chowdhury, I. J., & Nayak, R. (2014). *BEST: an efficient algorithm for mining frequent unordered embedded subtrees.* Paper presented at the Pacific Rim International Conference on Artificial Intelligence.

[13] Chowdhury, I. J., & Nayak, R. (2014). *BOSTER: an efficient algorithm for mining frequent unordered induced subtrees.* Paper presented at the International Conference on Web Information Systems Engineering.

[14] Davenport, T. H., & Short, J. E. (1990). The new industrial engineering: information technology and business process redesign.

[15] Deepak, A., Fernández-Baca, D., Tirthapura, S., Sanderson, M. J., & McMahon, M. M. (2014). EvoMiner: frequent subtree mining in phylogenetic databases. *Knowledge and Information Systems, 41*(3), 559-590.

[16] Guha, S., Grover, V., Kettinger, W. J., & Teng, J. T. (1997). Business process change and organizational performance: exploring an antecedent model. *Journal of management information systems, 14*(1), 119-154.

[17] Hadzic, F., Tan, H., & Dillon, T. S. (2007). *UNI3-efficient algorithm for mining unordered induced subtrees using TMG candidate generation.* Paper presented at the 2007 IEEE Symposium on Computational Intelligence and Data Mining.

[18] Hakawati, M. R., Yacob, Y., Raof, R. A. A., Jabiry, M. M. K., & Alhudiani, E. S. (2020). Data Cleaning Model for XML Datasets using Conditional Dependencies. *European Journal of Electrical Engineering and Computer Science, 4*(1).

[19] Hido, S., & Kawano, H. (2005). *AMIOT: induced ordered tree mining in tree-structured databases.* Paper presented at the Fifth IEEE International Conference on Data Mining (ICDM'05).

[20] Janes, A., Maggi, F. M., Marrella, A., & Montali, M. (2017). *From Zero to Hero: A Process Mining Tutorial.* Paper presented at the International Conference on Product-Focused Software Process Improvement.

[21] Jiménez, A., Berzal, F., & Cubero, J.-C. (2010). POTMiner: mining ordered, unordered, and partially-ordered trees. *Knowledge and information systems, 23*(2), 199-224.

[22] Kim, K., Yeon, M., Jeong, B.-S., & Kim, K. P. (2017). A Conceptual Approach for Discovering Proportions of Disjunctive Routing Patterns in a Business Process Model. *TIIS, 11*(2), 1148-1161.

[23] Li, Z., Xu, C., & Liu, C. (2019). Frequent Subtree Mining Algorithm for Ribonucleic Acid Topological Pattern. *Revue d'Intelligence Artificielle, 33*(1), 75-80.

[24] Lindsay, A., Downs, D., & Lunn, K. (2003). Business processes—attempts to find a definition. *Information and software technology, 45*(15), 1015-1019.

[25] Liu, W., & Chen, L. (2012). An efficient way of frequent embedded subtree mining on biological data. *J. Comput, 6,* 2574-2581.

[26] Mannhardt, F. (2016). XESLite-managing large XES event logs in ProM. *BPM Center Report BPM-16-04,* 224-236.

[27] Nijssen, S., & Kok, J. N. (2003). *Efficient discovery of frequent unordered trees.* Paper presented at the First international workshop on mining graphs, trees and sequences.

[28] Nijssen, S., & Kok, J. N. (2004). *A quickstart in frequent structure mining can make a difference.* Paper presented at the Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining.

[29] Romei, A., & Turini, F. (2010). XML data mining. *Software: Practice and Experience, 40*(2), 101-130.

[30] Rückert, U., & Kramer, S. (2004). *Frequent free tree discovery in graph data.* Paper presented at the Proceedings of the 2004 ACM symposium on Applied computing.

[31] Sadredini, E., Rahimi, R., Wang, K., & Skadron, K. (2017). *Frequent subtree mining on the automata processor: challenges and opportunities.* Paper presented at the Proceedings of the International Conference on Supercomputing.

[32] Shaharanee, I. N. M., & Jamil, J. M. (2015). Irrelevant feature and rule removal for structural associative classification. *Journal of Information and Communication Technology, 14*, 95-110.

[33] Shasha, D., Wang, J. T.-L., & Zhang, S. (2004). *Unordered tree mining with applications to phylogeny.* Paper presented at the Proceedings. 20th International Conference on Data Engineering.

[34] Singhal, S., & Jena, M. (2013). A study on WEKA tool for data preprocessing, classification and clustering. *International Journal of Innovative technology and exploring engineering (IJItee), 2*(6), 250-253.

[35] Studiawan, H., Sohel, F., & Payne, C. (2020). *Automatic event log abstraction to support forensic investigation.* Paper presented at the Proceedings of the Australasian Computer Science Week Multiconference.

[36] Tan, H., Dillon, T. S., Hadzic, F., Chang, E., & Feng, L. (2006). *IMB3-Miner: mining induced/embedded subtrees by constraining the level of embedding.* Paper presented at the Pacific-Asia Conference on Knowledge Discovery and Data Mining.

[37] Tatikonda, S., Parthasarathy, S., & Kurc, T. (2006). *TRIPS and TIDES: new algorithms for tree mining.* Paper presented at the Proceedings of the 15th ACM international conference on Information and knowledge management.

[38] Termier, A., Rousset, M.-C., & Sebag, M. (2002). *Treefinder: a first step towards xml data mining.* Paper presented at the 2002 IEEE International Conference on Data Mining, 2002. Proceedings.

[39] Tibeme, B., Shahriar, H., & Zhang, C. (2018). *Process Mining Algorithms for Clinical Workflow Analysis.* Paper presented at the SoutheastCon 2018.

[40] Trkman, P. (2010). The critical success factors of business process management. *International journal of information management, 30*(2), 125-134.

[41] Van der Aalst, W. M., van Dongen, B. F., Herbst, J., Maruster, L., Schimm, G., & Weijters, A. J. (2003). Workflow mining: A survey of issues and approaches. *Data & knowledge engineering, 47*(2), 237-267.

[42] van Dongen, B. F., & Van der Aalst, W. M. (2005). A Meta Model for Process Mining Data. *EMOI-INTEROP, 160*, 30.

[43] Wang, C., Hong, M., Pei, J., Zhou, H., Wang, W., & Shi, B. (2004). *Efficient pattern-growth methods for frequent tree pattern mining.* Paper presented at the Pacific-Asia conference on knowledge discovery and data mining.

[44] Weske, M. (2007). Business Process Management–Concepts, Languages, Architectures, Verlag. *Berlin*.

[45] Xiao, Y., & Yao, J.-F. (2003). *Efficient data mining for maximal frequent subtrees.* Paper presented at the Third IEEE International Conference on Data Mining.

[46] Zaki, M. J. (2005). Efficiently mining frequent embedded unordered trees. *Fundamenta Informaticae, 66*(1-2), 33-52.

[47] Zaki, M. J. (2005). Efficiently mining frequent trees in a forest: Algorithms and applications. *IEEE transactions on knowledge and data engineering, 17*(8), 1021-1035.

[48] Zhang, S., & Wang, J. T. (2007). Discovering frequent agreement subtrees from phylogenetic data. *IEEE Transactions on Knowledge and Data Engineering, 20*(1), 68-82.

Notes on contributors

Ang Jin Sheng is a PhD student at School of Quantitative Sciences, University Malaysia. He received his Master Studies in Federation University Australia in 2018. His research interest are data mining, artificial intelligence and big data analysis. He can be contacted at email: angjinsheng@gmail.com.

Dr. Jastini Mohd Jamil is a senior lecturer and researcher in Data Mining at School of Quantitative Sciences, Universiti Utara Malaysia. She received her Ph. D in Data Mining from University of Bradford in 2012. Her research interests are solving problems in diverse area using data mining, decision support system and statistical techniques. Her other interests include structural equation modeling, partial least squares, neural networks, rough sets, data pre-processing, handling missing data and forecasting. She can be contacted at email: jastini@uum.edu.my.

Associate Prof. Dr. Izwan Nizal Mohd Shaharanee is a lecturer and researcher at the Department of Decision Science, School of Quantitative Sciences, Universiti Utara Malaysia. He received his PhD from Curtin University, Perth, Australia in 2012. His research areas mainly focus on data mining especially the quality issues, measures of interestingness, evaluation and application of data mining models. He also has a great interest in solving problems in diverse area using data mining, decision support system and statistical techniques. He can be contacted at email: nizal@uum.edu.my