International Journal of Electrical and Computer Engineering (IJECE)
Vol. 9, No. 6, December 2019, pp. 5446~5453
ISSN: 2088-8708, DOI: 10.11591/ijece.v9i6.pp5446-5453

Analysis study on R-Eclat algorithm in infrequent itemsets mining

Mustafa Man[1], Julaily Aida Jusoh[2], Syarilla Iryani Ahmad Saany[3], Wan Aezwani Wan Abu Bakar[4], Mohd Hafizuddin Ibrahim[5]
[1]School of Informatics and Applied Mathematics, Universiti Malaysia Terengganu, Malaysia
[2,3,4]Faculty of Informatics and Computing, Universiti Sultan Zainal Abidin, Malaysia
[5]Department of Electrical Engineering, Politeknik Kuala Terengganu, Malaysia

Article Info

Article history:

Received Jan 10, 2019
Revised Apr 18, 2019
Accepted Jun 10, 2019

Keywords:

Data mining
Eclat algorithm
Infrequent itemset
Itemset mining
R-Eclat algorithm

ABSTRACT

There are rising interests in developing techniques for data mining. One of the important subfield in data mining is itemset mining, which consists of discovering appealing and useful patterns in transaction databases. In a big data environment, the problem of mining infrequent itemsets becomes more complicated when dealing with a huge dataset. Infrequent itemsets mining may provide valuable information in the knowledge mining process. The current basic algorithms that widely implemented in infrequent itemset mining are derived from Apriori and FP-Growth. The use of Eclat-based in infrequent itemset mining has not yet been extensively exploited. This paper addresses the discovery of infrequent itemsets mining from the transactional database based on Eclat algorithm. To address this issue, the minimum support measure is defined as a weighted frequency of occurrence of an itemsets in the analysed data. Preliminary experimental results illustrate that Eclat-based algorithm is more efficient in mining dense data as compared to sparse data.

Corresponding Author:

Julaily Aida Jusoh,
Faculty of Informatics and Computing,
Universiti Sultan Zainal Abidin,
Tembila Campus, 22200 Besut, Terengganu, Malaysia.
Email: julaily@unisza.edu.my

1. INTRODUCTION

Big data concerns the large volume and complex structured and unstructured data [1, 2]. The primary key of big data is to obtain valuable information or knowledge for future action. The information or knowledge mining process has to be very efficient and speedy in runtime during the process of storing all observed data. Big data possess many challenging issues of data mining and information processing. For most of the applications such as e-commerce, industry, and medicine, the challenge is to discover and extract valuable knowledge from big data for prediction services support.

Data mining [3, 4] plays an essential role in big data solutions since it can extract valuable knowledge from the complex systems. It is a subfield of computer science which blends many techniques from statistics, data science, database theory, and machine learning. The objective of data mining is to predict the future or to understand the past. Prediction is essential to estimate future work by analyzing the existing data. However, several data mining techniques aim at discovering patterns. Approaches for identifying patterns in data can be classified by the types of patterns that they discover. Some common types of patterns found in databases are clusters, itemsets, trends, and outliers.

Itemset mining [3-5] is one of the well-known tasks in discovering valuable correlations among data. There are two types of itemset mining that usually can be found in a database such as frequent and

infrequent. Frequent itemset mining focuses on patterns that frequently occurred, while the infrequent itemset mining, highlights on patterns that rarely occurred. Usually, the itemset mining algorithms target the extraction of frequent itemsets that have a high frequency in the transactional database. However, the infrequent itemsets with low support also can produce potential vital association rules among itemsets. It may contribute a significantly reliable decision support system. In the infrequent itemset mining, the first stage is to find out the infrequent itemsets from the transaction database. Followed by the second stage is to search out the association rules according to infrequent itemsets.

Any algorithm shall find the same set of rules, although their computational efficiencies and memory requirements may be different. The two basic algorithms of infrequent mining are Apriori and FP-Growth. Two types of data layouts which are usually employed in itemset mining to represent databases. They are the horizontal and vertical layout. In the horizontal database layout, each transaction consists of a set of items whereby the database is a set of transactions. Most Apriori-based algorithms employ this type of layout while FP-Growth uses both data layout for the mining purpose.

In this article, analysis performance on the infrequent mining using the R-Eclat algorithm will be discussed. Each variant of R-Eclat algorithm has a different work-flow. Thus, through experimentation, the difference of achievement in running time allows the determination of the fastest R-Eclat variant in mining infrequent itemset. According to the experimentation, IF-Diffset demonstrates the fastest performance with the lowest running time during the mining processing in both sparse and dense data.

2. RESEARCH METHOD

In the past few years, infrequent mining has made a leap in data mining. Infrequent patterns usually applied in diverse areas comprising biology, medicine, and security. In the medical domain, by analysing clinical databases of patients' diseases, the discovered infrequent patterns or trends will assist the medical officer to make decisions about the medicine prescribe or clinical care.

This paper addresses the problem of mining infrequent itemsets from transactional datasets. Let itemset $I = \{i_1, i_2, \ldots, i_m\}$ be a set of data items. More specifically, k-itemset denotes as a set of k items in I. A transactional dataset $T = \{t_1, t_2, \ldots, t_n\}$ is a set of transactions, where each transaction t_q ($q \in [1, n]$) is a set of items in I and is characterized by a transaction ID (tid). The support (frequency of occurrence of an itemset) of an itemset is the number of transactions containing I in T. An itemset I is infrequent if its support is less than or equal to a minimum support threshold. Otherwise, it is called to be frequent. Given a transactional dataset T and a minimum support threshold, the infrequent itemset mining problem involves determining all infrequent itemsets from T.

Apriori [6, 7] algorithm is the most commonly itemset mining algorithms that uses a breadth-first search and the downward closure property. It is usually adopted horizontal layout to represent the transaction database and the frequency of an itemset is computed by counting its occurrence in each transaction. Apriori discovered rules by exploiting support and confidence requirements and using the threshold to prune the search space. But, it is not efficient to find a low-support rules. Using Apriori, it needs to wade through thousands of itemsets (often having high support) to find the infrequent itemsets.

Another algorithm is to use a tree-based approach. Most tree-based infrequent pattern mining approaches follow the traditional FP-Growth algorithm [8]. FP-Growth [4] employs a divide and conquer strategy and a FP-tree data structure to achieve a condensed representation of the transaction database. It is a two-pass approach and is only affordable when mining a static dataset.

Liu et al. [9] proposed Multiple Support Apriori (MSApriori) to deal with the infrequent itemset by using multiple minimum support in a transactional database. In their research premise, they note that several itemset are rarely occurs in dataset. They cannot contribute to rules generated by Apriori, even though they may participate in rules that have a very high confidence. They overcome this problem with a technique whereby each item in the database can have its own minimum item support (MIS). By providing a different MIS for different items, a higher minsup can be set for rules apply on frequent items and lower minsup is rules that include infrequent items. Rahman [10] has designed the Online Apriori-Infrequent algorithm. This algorithm considers the support value but does not use the confidence value. It efficiently uses support to compute an anomaly score for the record. It determines whether this record is anomalous or not on the fly. An anomaly score is assigned to each packet (record) based on whether the record has more frequent or infrequent patterns. This algorithm improves the join and prune step of the traditional Apriori algorithm with a constraint. The constraint avoids joining itemsets not likely to produce frequent itemsets as their results, thereby improving efficiency and run times significantly.

Haglin and Manning [5] introduced an algorithm known as MINIT (Minimally Infrequent Item set) for mining minimal infrequent items by sorting and ranking items based on the support. The items that hold support less than minsup are selected. Then, only the transactions which contain those items are chosen for

further processing. A ranking list of items is prepared by calculating the support of each items and then created a list of items in ascending order of support. MINIT is suitable for a small size of dense dataset. Extension to MINIT, Gupta [11] has proposed a technique called minimally infrequent itemsets that is also mentioned as MII.

MII This technique is designed based on Inverse FP-Tree (IFP). The IFP-Tree recursively mines minimally infrequent itemsets by separating the IFP-tree into two subtrees called as projected and residual. The projected database corresponds to the set of transactions that contains a particular item. A potential minimal infrequent itemset mined from the projected tree must not have any infrequent subset. This is because the itemset itself is a subset. A residual tree for a particular item is a tree representation of the residual database corresponding to the item. The use of residual trees reduces the computation time and suitable for large dense datasets.

The RP - Tree algorithm was developed by Tsang et al. [12] to avoid the expensive itemset generation and pruning steps by using a tree data structure. It is designed based on FP-Tree and utilized a two-pass approach to find the infrequent patterns. RP-Tree executes a database scan to count the item support. Then, during the second scan, only the transactions which include at least one rare item will be used to build the initial tree and prunes the others. The proposed RP-Tree algorithm is an improvement over these existing algorithms in three ways as follows:

- RP-Tree avoids the expensive itemset generation and pruning steps by using a tree data structure, based on FP-Tree, to find rare patterns.
- RP-Tree focuses on rare-item itemsets which generates interesting rules and does not consume much time looking for uninteresting non-rare-item itemsets.
- RP-Tree is based on FP-Growth, which is efficient at finding long patterns, since the task is divided into a series of searches for short patterns. This is especially beneficial since infrequent patterns tend to yield longer path than frequent patterns.

Generally, Apriori and tree-based algorithms are developed to find infrequent itemsets. Tree-based approaches are more efficient as they do not require a candidate generation and multiple data scanning as Apriori. However, building the tree for rare pattern mining is more complicated. Recently, Jusoh et.al [13, 14] proposes a vertical data representation, called R-Eclat algorithm, an Eclat-based approaches using intersecting transaction ID. Eclat-based [15] takes a depth-first search and adopts a vertical layout to represent databases, in which each item is represented by a set of transaction IDs (also called as tidsets).

The remainder of this paper is organized as follows: Section 3 discusses the R-Eclat algorithm in details. The results of the performance analysis will be presented in Section 4. Section 5 sums up the paper with conclusions.

3. R-ECLAT ALGORITHM

In [15], Zaki proposes Equivalence Class Transformation (ECLAT) algorithm, which transforms the original dataset into a vertical database format. Each single item is stored in the dataset together with a list of transaction-ids (tidset) where the item can be found. ECLAT considers the frequency of a pattern P as the length of the transaction-ids list. This algorithm determines that any itemset is frequent if it lists at least f_{min} transaction-ids, i.e. $|t(P)| \geq f_{min}$. An important feature of ECLAT is the fast intersecting tids list, thus the size of tids list is one of the main factors affecting the execution time and memory usage of ECLAT. Nevertheless, its execution time is less than Apriori algorithm. ECLAT is the first algorithm that uses a vertical data layout and very efficient for large itemsets but less efficient for small itemsets. The frequent itemsets are determined using simple tids list intersections in a depth-first search graph. A variation of the Eclat algorithm that is implemented using different structure are called tidset (Eclat) [15], diffset [16], sortdiffset [17] and iEclat [13, 18]. Motivating from the fast intersections of tids list in ECLAT as well as how does it affects the running time and memory usage, it brings the author to explore and discover a further research in this algorithm especially for mining infrequent patterns.

The R-ECLAT algorithm is a novel technique that specially generated for infrequent patten mining [14, 19]. It is designed based on the traditional ECLAT algorithm. This algorithm uses a depth-first search to achieve a condensed representation of the transaction database. It utilizes column-based (vertical) rather than row-based (horizontal) to represent the dataset. R-ECLAT counts the support through determining support of any k-itemsets on the intersecting tid-lists of its k-1 subsets. In the R-ECLAT algorithm, the traditional tidset, diffset and sort-diffset variants are enhanced to ensure that it is appropriate for mining an infrequent pattern. The Figure 1 illustrates the model of R-Eclat division formats which are named as IF-Tidset, IF-Diffset and IF-Sortdiffset where IF is represented as infrequent.

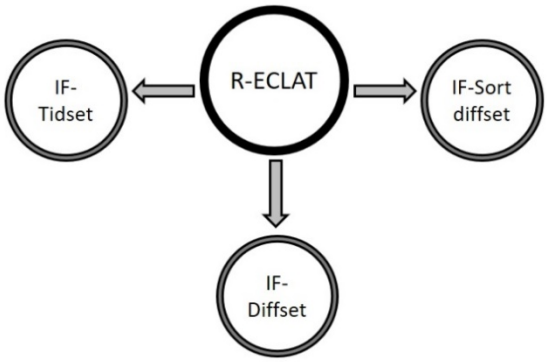

Figure 1. R-Eclat algorithm model [16]

The main steps in R-Eclat over the dataset are listed as follows:
Step 1 (Generation): scans the database to generate k-itemset candidates from two frequent (k-1)-itemsets and its support is counted.
Step 2 (Prune): if its support is greater than the minimum support threshold, then it will be discarded, otherwise it is denoted as infrequent itemsets and used to generate (k+1)-itemsets.

Step 1 is repeated until no candidate itemset can be generated. Subsequently, the minimum support threshold value (MSTV) is considered as a benchmark to discover a low occurrence in each dataset. In [18], MSTV is determined in terms of the percentage,

$$\frac{\delta}{100} * \alpha$$

where
δ = User specified minimum support value
α = Total of records in datasets.
In each loop, starts with the first loop, if the support is less than or equal (<=) to min_supp, then,

- In IF-Tidset, obtain the result of intersection between i^{th} column and i^{th}+1 column and save to the database
- In IF-Diffset, instead of using intersection, it acquires the result of diffset (difference intersection set) between i^{th} column and i^{th}+1column and saves the result to the database
- In IF-Sortdiffset, itemsets are first sorted in descending order which depends upon the highest to lowest value of itemset's equivalence class. Then the diffset value between i^{th} column and i^{th}+1 column will be encountered and saved to the database.

The detail experimentation regarding the IF-Tidset, IF-Diffset and IF-Sortdiffset format via different data characteristics are presented in the next section.

4. RESULTS AND DISCUSSION

All experiments are performed on two different processors: - 1) LENOVO Ideapad, Intel ® Core ™ ® i5-4210U CPU @ 2.40 GHz with 8GB RAM in a Win10 64-bit platform and 2) HP Notepad, Intel ® Core ™ ® i7-3520M CPU @ 2.90 GHz with 8GB RAM, in a Win10 64-bit platform. The raw benchmark datasets are retrieved from Frequent Itemset Mining Dataset Repository (*http://fimi.ua.ac.be/data/*) in a *.dat file format. For the ease of the uses, the selected benchmark data sets are transformed to Structured Query Language (SQL) format. All R-ECLAT variant algorithm formats are implemented in PHP programming. For the experimentation purposes, the datasets are first 'cleaned', where the instances consist of incomplete data and attributes of only one categorical value is removed. In this experiment, a thousand rows of itemsets have been randomly processed for mining purposes.

In order to evaluate the performance of the R-ECLAT algorithm, four datasets are used. They are datasets chess, retails, mushroom and T40I10D100K. Table 1 depicts the characteristics of the chosen datasets. For the experimental purpose, the minimum support threshold value is set at 3%.

Table 1. Datasets characteristics

Datasets	Records (Transaction)	Length (Attribute)	Size (KB)	Data Characteristic
Chess	3196	37	335	Dense
Mushroom	8125	43	558	Dense
Retails	88162	68	5143	Sparse
T40I10D100K	100001	32	15116	Sparse

The experimentation involves all R-ECLAT algorithm variants consist of IF-Tidset, IF-Diffset and IF-Sortdiffset. The performance between two processors (Intel core i5 and Intel core i7) is measured in terms of its runtime in both dense and sparse datasets. Figure 2 until Figure 5 shows the performance evaluation on execution time of each datasets for the R-ECLAT via serial processing.

Based on the observation of performance analysis for both processors as illustrated in Figure 2 to Figure 5, the runtime via Intel core i5 seems to drastically lose its performance over Intel core i7 in both dense and sparse datasets. In those datasets experimentation, IF-Diffset is extensively outperforms in mining infrequent itemsets with the lowest execution times as compared to IF-Tidset and IF-Sortdiffset. The results also illustrate that the next best performance are IF-Sortdiffset and IF-Tidset respectively. The result of these experiments can be concluded that there are two factors contribute to the overall performance of the infrequent itemset mining. First, the processor architecture and the second is the nature of datasets in terms of how many times the occurrence of itemsets. Nevertheless, in serial processing, different speed of processors does not show any significantly impact in mining the data. The experimental result only shows a slightly difference between Intel Core i5 and Intel Core i7 which are in average of 6% in sparse dataset and 12% in dense data. This may suggest that the parallel processing approach is relatively surpass the performance of a serial processing.

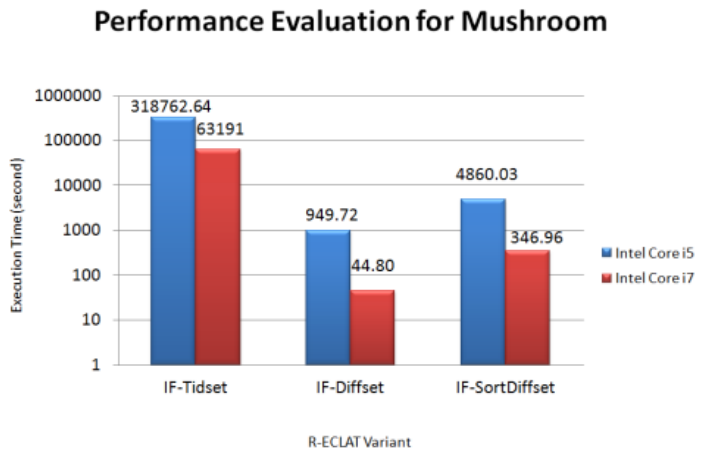

Figure 2. Intel core i5 vs Intel core i7: performance evaluation for mushroom

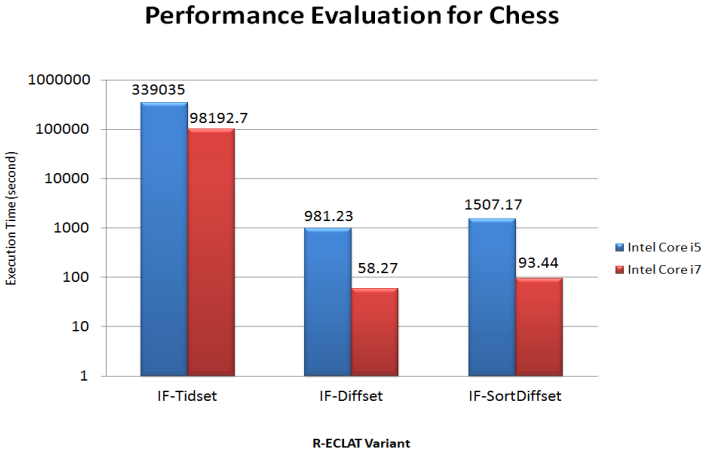

Figure 3. Intel core i5 vs Intel core i7: performance evaluation for chess

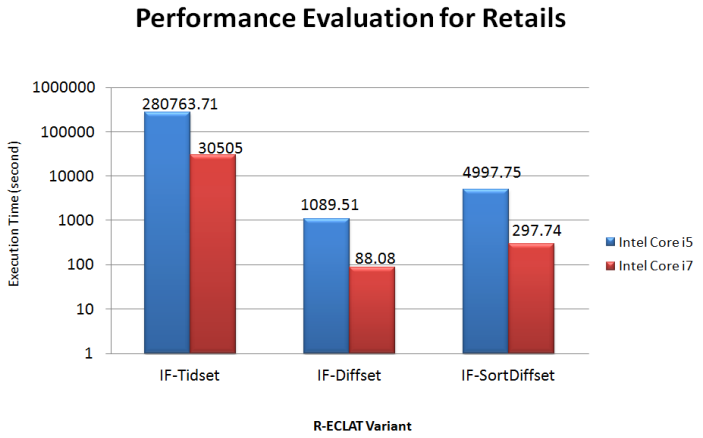

Figure 4. Intel core i5 vs Intel core i7: performance evaluation for retails

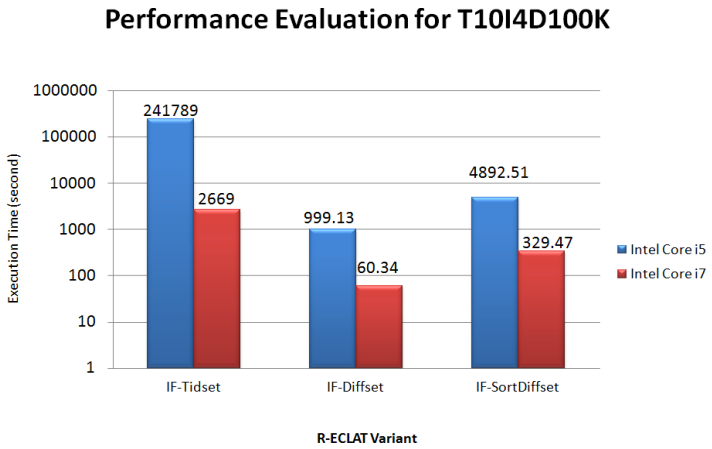

Figure 5. Intel core i5 vs Intel core i7: performance evaluation for T10I4D100K

5. CONCLUSION

In this paper, R-Eclat is presented as a potential solution in the infrequent itemset mining issues as mentioned in the earlier section. The experimentation results show that the data characteristic can greatly impact the running time. There are many other measurement parameters that can be imposed on the R-ECLAT algorithm to demostrate either the performance result between its variants remain the same or otherwise. In support measure, the IF-Diffset is found to be a better algorithm format in encountering the infrequent itemsets of the transactional database. The experimentation results also show that the data characteristic can greatly impact the running time. At present, R-Eclat is still undergoing a continuous enhancement in accelerating the mining process of infrequent itemset in order to be a preferred solution for a parallel or serial processing.

ACKNOWLEDGMENT

We express our gratitude to MyPhD scholarship under SLAB of Kementerian Pendidikan Malaysia (KPM) and Fundamental Research Grant Scheme (FRGS), Vot Grant: 59543 for the financial support for this work.

REFERENCES

[1] Y. Djenouri, *et al.*, "How to Exploit High Performance Computing in Population-Based Metaheuristics for Solving Association Rule Mining Problem," Distributed *Parallel Databases*, vol. 36, 2018.
[2] I. Yacoob, *et al.*, "Big Data: From Beginning to Future," *International Journal of Information Management*, vol. 36, 2016.
[3] C. C. Aggarwal, *"Data Mining: The Textbook,"* Heidelberg, Springer, 2015.

[4] J. Han, *et al.*, "Data Mining: Concepts and Techniques," Amsterdam, Elsevier, 2011.
[5] D. J. Haglin and A. M. Manning, *"On Minimal Infrequent Itemset Mining,"* Proceedings of the International Conference on Data Mining, *DMIN'07,* CSREA Press, pp. 141-147, 2007.
[6] R. Agrawal, *et al.*, "Mining Association Rules Between Sets of Items in Large Databases," *ACM SIGMOD*, 1993.
[7] R. Agrawal and R. Srikant, *"Fast Algorithms for Mining Association Rules in Large Databases,"* Proceedings of the 20th International Conference on Very Large Data Bases (VLDB), Santiago de Chile, Chile, pp. 487-499, 1994.
[8] J. Han, *et al.*, "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach," *Data Mining and Knowledge Discovery*, vol. 8, pp.53-87, 2004.
[9] B. Liu, *et al.*, "Mining Association Rules with Multiple Minimum Supports," *Proceedings of the 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 337-341, 1999.
[10] A. Rahman, *et al.*, "*Wifi Miner: An Online Apriori-Infrequent Based Wireless Intrusion System*," Knowledge Discovery from Sensor Data, Lecture Notes in Computer Science, Springer, Berlin, vol. 5840, pp.76-93, 2010.
[11] A. Gupta, *et al.*, "Minimally Infrequent Itemset Mining Using Pattern-Growth Paradigm and Residual Trees," CoRR abs/1207.4958, 2012.
[12] S.Tsang, *et al.*, *"RP-tree: Rare Pattern Tree Mining," DaWaK*, *Lecture Notes in Computer Science*, Alfredo Cuzzocrea and Umeshwar Dayal (Eds.), Springer, Berlin, vol. 6862, pp. 277-288, 2011.
[13] J. A. Jusoh and M. Man, "Modifying iEclat Algorithm for Infrequent Patterns Mining," *Advanced Science Letters*, vol. 24, 2018.
[14] J. A. Jusoh, *et al.*, "Mining Infrequent Patterns Using R-Eclat Algorithms," *Journal of Fundamental and Applied Sciences*, vol. 24, 2018.
[15] M. J. Zaki, *et al.*, "*New Algorithms for Fast Discovery of Association Rules,"* ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 283-286, 1997.
[16] M. J. Zaki and K. Gouda, *"Fast Vertical Mining Using Diffsets,"* ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003.
[17] T. A. Trieu and Y. Kunieda, "*An Improvement for Declat Algorithm*," The 6th International Conference on Ubiquitous Information Management and Communication, vol. 54, 2012.
[18] W. A. B. W. A. Bakar, *et al.*, *"Incremental-Eclat Model: An Implementation via Benchmark Case Study,"* Springer International Publishing Switzerland, P.J. Soh et al. (eds.), Advances in Machine Learning and Signal Processing, Lecture Notes in Electrical Engineering, vol. 387, pp. 35-46, 2016.
[19] M. Man, *et al.*, "Postdifset Algorithm in Rare Pattern: An Implementation via Benchmark Case Study," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 8, pp. 4477-4485, 2018.

BIOGRAPHIES OF AUTHORS

Mustafa Man is an Associate Professor in School of Informatics and Applied Mathematics and also as a Deputy Director at Research Management Innovation Centre (RMIC), UMT. He started his PhD studies in July 2009 and finished his studies in Computer Science from UTM in 2012. He has received Computer Science Diploma, Computer Science Degree, Masters Degree from UPM. In 2012, he has been awarded a "MIec MOS Prestigious Awards" for his PhD by MIMOS Berhad. His research is focused on the development of multiple types of databases integration model and also in Augmented Reality (AR), android based, and IT related into across domain platform.

Julaily Aida Jusoh received her B.Eng (Hons) in Software Engineering from the Universiti Putra Malaysia (UPM), Selangor in 2004. After graduated, she furthered her Master study in Software Engineering in Universiti Malaysia Terengganu (UMT) in 2005. In 2009, she joined Universiti Sultan Zainal Abidin (UNISZA) as a lecturer. Now, she furthered her PhD studies in Universiti Malaysia Terengganu (UMT) since September 2016. She currently works in infrequent itemset mining using Eclat Algorithm for her PhD research. Her current research interests include software engineering, formal methods and itemset mining.

Syarilla Iryani Ahmad Saany received her Bachelor of Science degree in Computer Information System at California State University, Chico, United States in 1997. She joined Universiti Sultan Zainal Abidin (formerly known as Sultan Zainal Abidin Islamic College) as a lecturer in 1997. She was awarded a scholarship and enrolled in the Master program at Universiti Kebangsaan Malaysia. She obtained her Master in Computer Science in 2005. She received her PhD in Intellegence Computing from Universiti Putra Malaysia in May 2015. Now, she is an Associate Professor in Faculty of Informatics and Computing. Her research interests are knowledge management, semantics and, e-Learning.

Wan Aezwani Wan Abu Bakar received her PhD in Computer Science at Universiti Malaysia Terengganu (UMT) Terengganu in Nov, 2016. Her focus area is in association rule in frequent itemset mining. She received her master's degree in Master of Science (Computer Science) from Universiti Teknologi Malaysia (UTM) Skudai, Johor in 2000 prior to finishing her study in Bachelor's degree also in the same stream from Universiti Putra Malaysia (UPM) Serdang, Selangor in 1998. Her master's research was formerly on Fingerprint Image Segmentation in the stream of Image Processing. Now she's pursuing her research towards association relationship in infrequent itemset mining which is more downstream to educational data settings.

Mohd Hafizuddin Ibrahim has received Bachelor of Electrical Engineering (Hons) in 2007 and Diploma in Electrical Engineering with Technology in 2004 from UTHM. In 2007. After graduated he join the MDIENT Engineering as an Electrical Engineer in the shipbuilding industry. In 2009, he joined Politeknik Kuala Terengganu as a lecturer. Now he is Head of Program in Diploma in Electrical & Electronic Engineering at the Department of Electrical Engineering, Politeknik Kuala Terengganu (PKT). His research interests are in the areas of robotics, with focus on autonomous systems, sensor fusion and robot control.