IJCSE International Journal of Computer Sciences and Engineering Open Access

Research Paper Vol.-7, Issue-4, April 2019 E-ISSN: 2347-2693

Utility Association Rule Mining – A Comprehensive Study

C. Sivamathi[1*], S. Vijayarani[2]

[1,2]Department of Computer Science, Bharathiar University, Coimbatore,

Corresponding Author: c.sivamathi@gmail.com

DOI: https://doi.org/10.26438/ijcse/v7i4.206210 | Available online at: www.ijcseonline.org

Accepted: 18/Apr/2019, Published: 30/Apr/2019

Abstract: Utility mining is gaining attention towards researchers, as it discovers semantic significance among the items in a database. Utility association rule mining is one of utility mining techniques that retrieves highly profitable and highly associated products in a database. Many researchers started to replace traditional association rule mining with utility association rule mining, since utility association rules can reflect both association and semantic significance among the products retrieved from the database. Utility based association rule mining can be applied on various domains like Bio-informatics, Recommender systems, Medical database, Web mining, Image mining. This research work aims to provide in depth study on utility based association rule mining. The work also illustrates the need for utility association rules, by providing drawbacks of traditional association rules. The work also lists existing utility association rules algorithms.

Keywords – *Data mining; Utility mining; High utility itemsets; High utility Association rule mining; Utility based association rules; utility confidence.*

I. INTRODUCTION

Data mining is extraction of implicit information and knowledge which are previously unknown and potentially useful [1][2]. Utility mining is one of the fields in data mining, which incorporates utility factor for all the items in a database. If the utility of itemsets is greater than minimum utility threshold, then the itemsets are defined as high utility itemsets. Utility mining is defined as extraction of high utility itemsets from a database. Utility itemset mining concepts were introduced based on frequent itemsets mining. In frequent itemset mining, only the occurrences of items are focused, not the semantic significance of items like weight, profit, quantity of items. In order to include these significant factors, utility mining concepts were introduced. Hence in utility mining the important factors of the items like quantity, profit of the items are incorporated. The following are some of definitions in utility mining [3][4][5].

Definition 1: The internal utility of an item i_p is the quantity of an item purchased by a customer in a transaction.
Definition 2: The external utility of an item i_p is unit profit of each item.
Definition 3: Utility function f is the product of internal and external utility and it is considered as utility function.
Definition 4: The utility of an item i_p in transaction T is the calculated using utility function. Utility of an item in a particular transaction = Product of its internal utility in that transaction and its external utility.
Definition 5: The utility of an itemset S in transaction T is defined as $u(S,T) = \sum u(ip,T), \forall ip \in S, S \subseteq T$.
Definition 6: The utility of itemset S in database DB is defined as, $u(S) = \sum u(S, T), \forall \; T \in DB, S \subseteq T$.
Definition 7: The utility of transaction T is defined as $u(T) = Pip \in T \; u(ip, T)$.
Definition 8: Itemset S is said to be *high utility itemset* if and only if $U(S) \geq Min\_Util$ where Min_Util is user specified minimum utility threshold.

II. ASSOCIATION RULE MINING

Association rule mining is one of the core concepts of data mining. This technique retrieves highly co-related items in a database [8] [9]. It is useful for finding highly associated items from a large database. It has wide applications in retail business like associated products analysis, cross selling analysis, analysis of purchase behaviour of customer etc. Other than retail/wholesale business association rule mining is also applied in E-Commerce, Web mining, sequence mining, Recommender systems, Bio-informatics, Image mining, Text Mining, Privacy preserving, health data etc.

Finding associated products in a large database has two phases [10] [11]. Frequent itemset generation is the first phase and discovery of association rules from frequent itemsets is a second phase. In frequent itemset generation phase, frequently occurring items in a database are retrieved. The itemset whose support value exceeds minimum threshold is referred as frequent itemsets. The support value of an itemset is nothing but the number of occurrence of itemsets in that database. Once, frequent items are found, the association rules are retrieved based on support and confidence values of rules. A rule for two itemsets A and B, of the form A→B, is known as association rule, if the support and confidence of the rule exceeds their minimum threshold respectively. Support of the rule is the ratio of number of transactions that contains both itemsets A and B to the total number of transactions in a database. Confidence of the rule is defined as the ratio of number of transactions that contains both itemsets A and B to the number of transactions that contains the first itemset A only.

III. DRAWBACKS OF TRADITIONAL ASSOCIATION RULE MINING

The association rule mining defined in the above section was purely based on support and confidence of the rule. This support and confidence measures are based on the number of occurrence of the items in a database. Just like frequent itemset mining that do not reflects the semantic importance of the items, this traditional association rules also not reflects the importance of items. It retrieves highly occurred co-related items. But in real life there may be some items which has more importance than its occurrence [12] [13].

Consider a retail store. Assume that the profit of 10kg of wheat is 50 INR and the profit of a pen is 2 INR. For example, consider that in a transaction database, pen occurs in 50 transactions and 10kg rice occurs in 5 transactions. The total profit of pen is 100 INR and the total profit of 10kg of wheat is 250 INR. As per frequent itemset mining pen has high frequency. But the total profit of wheat much greater than a pen, though it has least occurrences. Hence, traditional association rule mining cannot discover the association of most profitable itemsets. This is because frequent itemset mining does not consider the profit of an item, which is also highly important in decision making [14] [15].

IV. UTILITY ASSOCIATION RULE MINING

The traditional association rule will not reveal semantic implication towards the mining knowledge, because of support–confidence framework [16][17]. Hence utility mining with traditional association rule mining are fused together as utility based association rule mining. The objective of utility based association rule mining is to discover high utility association rules [3][4][5]. A rule X->Y is defined as high utility association rule [6][7], if X, Y are high utility itemsets, and the utility confidence of the rule is greater than minimum utility confidence threshold. The utility confidence (Uconf) of the rule X -> Y is defined as,

$$\mathrm{Uconf}(\mathrm{A} -> \mathrm{B}) = \frac{\mathrm{localutility(A,AUB\ )}}{\mathrm{U(A)}} \qquad (1)$$

Here, local utility(X, XUY) represents the local utility value of an item X in XUY and is defined as the sum of the utility values of the items X in all the transactions containing both X and Y. U(X) is local utility of X in a database. Steps in generating high utility association rules involves two phases: First phase is generation of high utility itemsets from a database and the second phase is discovery of high utility association rules from the high utility itemsets got in the first phase. A very little attention has paid in this area.

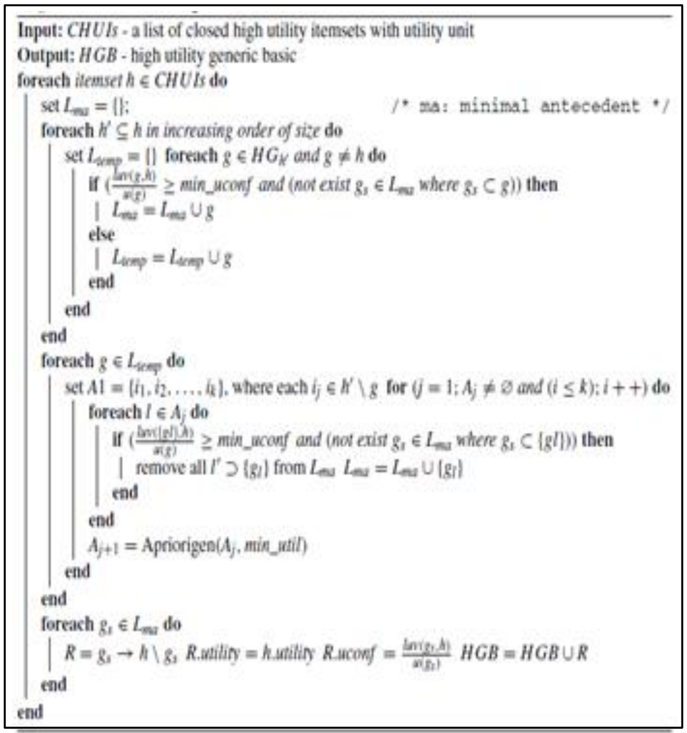
V. EXISTING UTILITY ASSOCIATION RULE MINING ALGORITHMS

A. HGB-HAR Algorithm

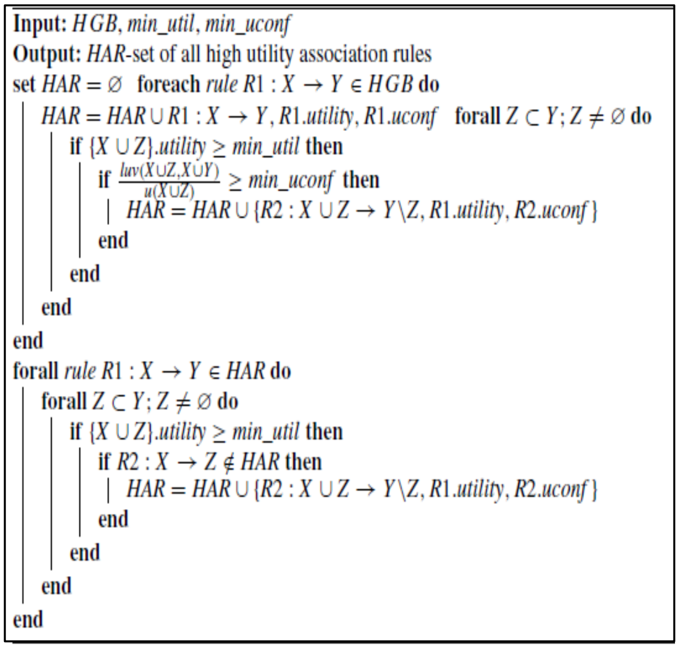
It was the first work on utility based association rules [6]. They defined utility confidence framework and differentiated how it differentiated from traditional association rule mining. The definition given in equation (1) was defined in this work [6]. The authors generated association rules from high utility closed itemsets (HUCI). An itemset is closed in a data set if there exists no superset that has the same support count as this original itemset. The authors proposed that high utility closed itemset together with their generators those are also high utility itemsets and also these high utility closed itemsets results in construction of effective non redundant association rules, as in traditional association rule mining. Hence they incorporated closed itemsets in utility itemsets and generate utility association rules from closed itemsets.

The authors proposed four procedures in this work. They constructed initial utility list to discover high utility itemsets. Then closed high utility items and its generators are discovered. From the resultant generators, high utility association rules are retrieved. The authors define these rules as non-redundant high utility association rules, as they are generated from closed items. This is because, closed items reduces redundancy in rule generation.

Step 1: Construct initial utility list for a transaction database.
Step 2: Discover a set of HUIs.
Step 3: Retrieve High utility closed itemsets (HUCIs) from HUIs.
Step 4: Generate High utility association rules



```
Input: $CHUIs$ - a list of closed high utility itemsets with utility unit
Output: $HGB$ - high utility generic basic
foreach itemset $h \in CHUIs$ do
    set $L_{ma} = \{\}$;                                 /* ma: minimal antecedent */
    foreach $h' \subseteq h$ in increasing order of size do
        set $L_{temp} = \{\}$ foreach $g \in HG_{h'}$ and $g \neq h$ do
            if ($\frac{luv(g,h)}{u(g)} \geq min\_uconf$ and (not exist $g_s \in L_{ma}$ where $g_s \subset g$)) then
                $L_{ma} = L_{ma} \cup g$
            else
                $L_{temp} = L_{temp} \cup g$
            end
        end
    end
    foreach $g \in L_{temp}$ do
        set $A1 = \{i_1, i_2, \ldots, i_k\}$, where each $i_j \in h' \setminus g$ for ($j = 1; A_j \neq \emptyset$ and $(i \leq k); i++$) do
            foreach $l \in A_j$ do
                if ($\frac{luv(\{gl\},h)}{u(g)} \geq min\_uconf$ and (not exist $g_s \in L_{ma}$ where $g_s \subset \{gl\}$)) then
                    remove all $l' \supset \{g_l\}$ from $L_{ma}$ $L_{ma} = L_{ma} \cup \{g_l\}$
                end
            end
            $A_{j+1} = \text{Apriorigen}(A_j, min\_util)$
        end
    end
    foreach $g_s \in L_{ma}$ do
        $R = g_s \rightarrow h \setminus g_s$ $R.utility = h.utility$ $R.uconf = \frac{luv(g_s,h)}{u(g_s)}$ $HGB = HGB \cup R$
    end
end
```



```
Input: $HGB, min\_util, min\_uconf$
Output: $HAR$-set of all high utility association rules
set $HAR = \emptyset$   foreach rule $R1 : X \rightarrow Y \in HGB$ do
    $HAR = HAR \cup R1 : X \rightarrow Y, R1.utility, R1.uconf$   forall $Z \subset Y; Z \neq \emptyset$ do
        if $\{X \cup Z\}.utility \geq min\_util$ then
            if $\frac{luv(X \cup Z, X \cup Y)}{u(X \cup Z)} \geq min\_uconf$ then
                $HAR = HAR \cup \{R2 : X \cup Z \rightarrow Y \setminus Z, R1.utility, R2.uconf\}$
            end
        end
    end
end
forall rule $R1 : X \rightarrow Y \in HAR$ do
    forall $Z \subset Y; Z \neq \emptyset$ do
        if $\{X \cup Z\}.utility \geq min\_util$ then
            if $R2 : X \rightarrow Z \notin HAR$ then
                $HAR = HAR \cup \{R2 : X \cup Z \rightarrow Y \setminus Z, R1.utility, R2.uconf\}$
            end
        end
    end
end
```

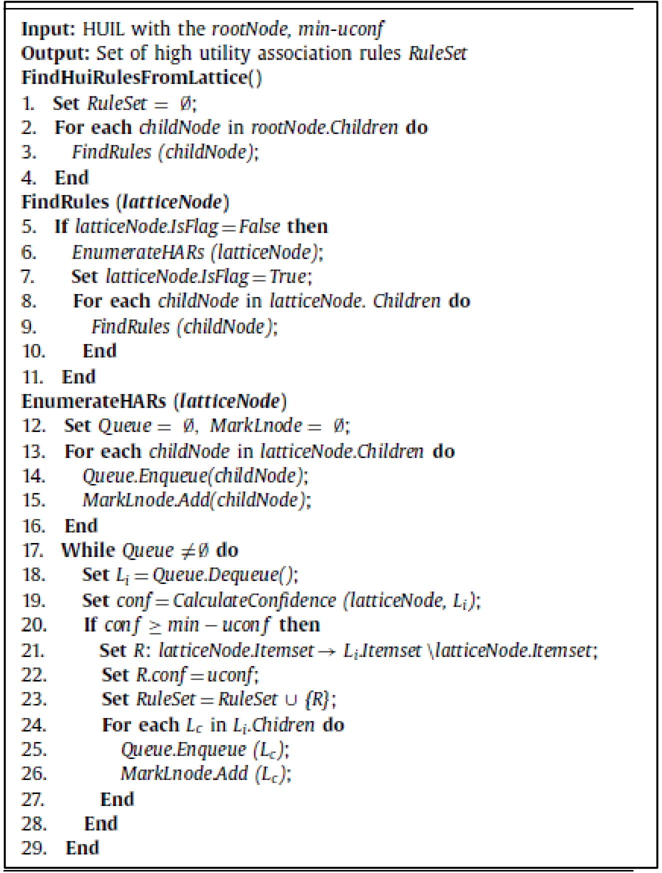
Pseudocode for HGB-HAR algorithm..(Image source [6])

B. LARM Algorithm

The authors argue that HGB-HAR algorithm has more memory and space complexity, if HGB list generated is very large [7]. Hence this algorithm was not more suitable for large databases. To overcome this limitation, the authors [7] proposed a lattice structure for mining high utility association rules. This work has two phases: In first phase a lattice for high utility itemsets (HUIL) was constructed. In second phase, all high utility association rules (HARs) are generated from the HUIL.

Step 1: Discover HUIS from a transaction database.
Step 2: Construct Lattice structure for HUIs.
Step 3: Generate High utility association rules

The nodes of items in lattice stores three data; {Utility of item, support of item, name of high utility itemset}. This lattice is built from the list of high utility itemsets. The lattice structure contains a root node and child nodes. The nodes are connected between each pairs of nodes. The root node is an empty node with no items and utility and support equal to 0. The connections between each pair of nodes are used to specify their parent-child relationship. The name of each node is formed based on the collection of items in an itemset. For example, if A is the root node, then the child nodes {AB, AC}.



```
Input: HUIL with the rootNode, min-uconf
Output: Set of high utility association rules RuleSet
FindHuiRulesFromLattice()
Set RuleSet = Ø;
For each childNode in rootNode.Children do
    FindRules (childNode);
End
FindRules (latticeNode)
If latticeNode.IsFlag = False then
    EnumerateHARs (latticeNode);
    Set latticeNode.IsFlag = True;
    For each childNode in latticeNode. Children do
        FindRules (childNode);
    End
End
EnumerateHARs (latticeNode)
Set Queue = Ø, MarkLnode = Ø;
For each childNode in latticeNode.Children do
    Queue.Enqueue(childNode);
    MarkLnode.Add(childNode);
End
While Queue ≠ Ø do
    Set $L_i$ = Queue.Dequeue();
    Set conf = CalculateConfidence (latticeNode, $L_i$);
    If conf ≥ min − uconf then
        Set R: latticeNode.Itemset → $L_i$.Itemset \latticeNode.Itemset;
        Set R.conf = uconf;
        Set RuleSet = RuleSet ∪ {R};
        For each $L_c$ in $L_i$.Chidren do
            Queue.Enqueue ($L_c$);
            MarkLnode.Add ($L_c$);
        End
    End
End
```

Pseudocode for LARM algorithm (Image source [7])

VI. EXAMPLE

Consider a transaction database as shown in table 1. First column is transaction ids. Second column is the items with their quantity purchased by the customer. Here T1, T2 T3, T4, T5, T6 are Transaction ids. A,B,C,D,E are items in transactions. The number along with the item in parentheses represents the quantity of item purchased. In utility mining, this quantity is termed as internal utility of item. Table 2 shows the unit profit of items. This unit profit is termed as external utility of items. The product of internal and external utility gives utility item in a transaction. Similarly Transaction utility is the sum of utility of items in those transactions. Transaction weighted utility is the sum of transaction utilities in which the item present. Table 2 shows unit profit , TWU and support of each items. More detailed definition about basics of utility mining can be found in [1][2][8][9][10].

Table 1. An Example transaction

Trans. Id	Transactions	TU
T1	(A,1) (B,3) (C,2)	19
T2	(A,2) (C,3) (D,1)	21
T3	(B,2) (C,1)	10
T4	(B,3) (C,1) (D,3) (E,2)	32
T5	(A,1) (B,3) (C,2) (D,1) (E,2)	28

Table2. Utility, TWU table

Item	Unit profit	TWU	Support
A	2	68	3
B	3	89	4
C	4	110	5
D	5	81	3
E	2	60	2

Initially TWU and support of items are calculated and is shown in table 3. Consider minimum utility threshold as 70 and minimum support as 3.Hence high utility items are A,B,C and D. Here A,B and C are closed high utility itemsets.

HGB-HAR Algorithm generates association rules from {A,B,C}. Once all possible rules are generated, the rules with utility value greater than utility threshold are termed as high utility association rules. LARM Algorithm constructs lattice for the items {A,B,C}. From the lattice the rules are retrieved.

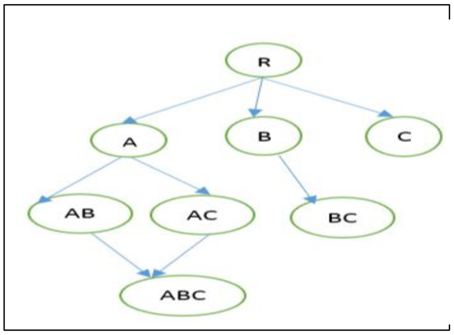


Fig 1. Itemsets Lattice of LARM Algorithm

Retrieved High utility association rules are {(A->B); (B->C), (A->BC), (AC->B)}

VII. CONCLUSION

Association rule mining techniques are used to discover the relationships between itemsets, based on support and confidence measures. The traditional association rule mining does not incorporate semantic association among itemsets. Hence to incorporate the semantic factors, utility association rules are introduced. The utility association rule introduces utility confidence for the rules, so that the associated products discovered through utility association rules reflects the semantic significance also. Moreover these rules are generated from high utility itemsets, hence the itemsets will implicitly has more significance than frequent itemsets. In this work a detailed study that includes Association Rule mining, drawbacks of traditional association rule, utility association rules are illustrated in detail. Also the work lists some of the existing utility association rule mining algorithms.

REFERENCES

[1] R. Agrawal , T. Imielinski , A. Swami , Mining association rules between sets of items in large databases, in: Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data, 1993, pp. 207–216 .

[2] R. Agrawal , R. Srikant , Fast algorithms for mining association rules, in: Proceedings of International Conference on Very Large Data Bases, VLDB'94, 1994, pp. 4 87–4 99 .

[3] Yao, H., Hamilton, H. J., & Geng, L. (2006). A unified framework for utility-based measures for mining itemsets. In Proceedings of ACM SIGKDD 2nd workshop on utility-based data mining (pp. 28–37). ACM.

[4] Chen, Y., Zhao, Y., & Yao, Y. (2007). A profit-based business model for evaluating rule interestingness. In Advances in artificial intelligence. Lecture notes in computer science (Vol. 4509, pp. 296–307). Berlin Heidelberg: Springer.

[5] Chan, R., Yang, Q., & Shen, Y.-D. (2003). Mining high utility itemsets. In Third IEEE international conference on data mining (ICDM 2003) (pp. 19–26). IEEE

[6] J. Sahoo , A.K. Das , A. Goswami , An efficient approach for mining association rules from high utility itemsets, Expert Syst. Appl. 42 (13) (2015) 5754–5778 .

[7] Mai, Thang, Bay Vo, and Loan TT Nguyen, A lattice-based approach for mining high utility association rules, Information Sciences 399 (2017): 81-97.

[8] Lee, D., Park, S. H., & Moon, S. (2013). Utility-based association rule mining: A marketing solution for cross-selling. Expert Systems with applications, 40(7), 2715-2725.

[9] Y. Liu , W. Liao , A. Choudhary ,A Two-Phase algorithm for fast discovery of high utility itemsets, in: Proceedings of the 9th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining, 2005, pp. 689–695 .

[10] V.S. Tseng , C. Wu , B. Shie , P.S. Yu , UP-Growth: an efficient algorithm for high utility itemset mining, in: Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2010, pp. 253–262 .

[11] Sivamathi, C., and S. Vijayarani. Performance analysis of utility mining algorithms. Inventive Computation Technologies (ICICT), International Conference on. Vol. 3. IEEE, 2016.

[12] Vijayarani,S., Sivamathi. C , Jeevika Tharani .V. Recent Trends in Utility Mining: A Survey. International Journal of Research in Information Technology.

[13] KannimuthuP, S., PremalathaP, K., & UshaP, G. Survey of recent developments in utility based Data mining.

[14] U. Yun , H. Ryang , K.H. Ryu ,High utility itemset mining with techniques for reducing overestimated utilities and pruning candidates, Expert Syst. Appl. 41 (8) (2014) 3861–3878 .

[15] M.J. Zaki , C.J. Hsiao , Efficient algorithms for mining closed itemsets and their lattice structure, IEEE Trans. Knowl. Data Eng. 17 (4) (2005) 462–478 .

[16] S. Zida, P. Fournier-Viger, J.W. Lin, C. Wu, V.S. Tseng, EFIM: a fast and memory efficient algorithm for high-utility itemset mining, Knowl. Inf. Syst. (2016) 1–31, doi: 10.1007/s10115- 016- 0986-0.

[17] Pramod Pardeshi, Ujwala Patil, "Fuzzy Association Rule Mining- A Survey", International Journal of Scientific Research in Computer Science and Engineering, Vol.5, Issue.6, pp.13-18, 2017

AUTHORS PROFILE

Mrs. C.Sivamathi, is pursing her Ph.D in Department of Computer Science, Bharathiar University, Coimbatore, Tamilnadu, India. She has completed M.Sc(CS & IT) and M.Phil(CS) in Madurai Kamaraj University, Tamilnadu, India. Her research area includes Utility Mining, Privacy Preserving and Optimization techniques. She has published papers in international journals and conferences.

Dr. S.Vijayarani Mohan is an Assistant Professor of Department of Computer Science at Bharathiar University, Coimbatore, India. She has obtained M.C.A., M.Phil., and Ph.D., in Computer Science. She has 10 years of teaching/research and 10 years of technical experience. Her research interests include data mining, privacy issues in data mining, text mining, web mining, data streams and information retrieval. She has published more than 100 research articles in national/international journals. She also presented research papers in international/national conferences. She has authored a book and guided more than 25 research scholars. She is a life member in professional bodies like CSI, ISCA, IAENG, IRED and UACEE.