Available online www.jsaer.com

***Journal of Scientific and Engineering Research*, 2021, 8(5):65-74**

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Efficient Method of Mining Sequential pattern in retail database

Oyekunle, Victoria Oluwatoyin[1], Nwanyanwu Mercy[2], Ide, Mercy Azibaye[3]

[1]Federal Polytechnic of Oil and Gas, Bonny, Rivers State, Nigeria
[2]Department of Computer Science, Captain Elechi Amadi Polytechnic Rumuola, Port Harcourt, Rivers State, Nigeria
[3]International Institute of Tourism and Hospitality Yenagoa, Bayelsa State, Nigeria
Email: [1]oluwatoyinoyekunle1@gmail.com, [2]mercynthia201@gmail.com, [3]arikawei_4real@yahoo.com

Abstract There is difficulty in detecting repetitive behavior for a company in order to identify regularities in businesses and inability to determine meta patterns as a group of events that lead to particular deviations in customers' behavior. However, a sequential pattern mining mechanism has been developed to discover all sequences and subsequences that are repetitive in a meta data. Percussive method of pattern growth algorithms of Sequential Pattern Mining (SPM) has been used for finding frequent patterns from a huge data set. It first scans the sequence database and calculate support of each data and find all frequent patterns which have support greater than support threshold, then sequence database converted into compressed data structure by removing all infrequent item sets. This process continues until all frequent pattern are generated. The protocol we propose for analyses are UK and Ireland dataset to determine the most frequent purchased product. The analysis requires that all the data for a transaction sequence be included in 1 row (like a receipt), which includes one receipt number and also the groceries list. The result shows items with a strong relationship; the market analysis could be used for the optimization of retail strategy.

Keywords Sequential Pattern Mining, Percussive Method, Retail Database

Introduction

As patterns from a sequence database, sequential pattern mining is one of the key areas of research in the field of data mining. With vast volumes of data being generated and stored on a continual basis, several companies are growing interest in extracting sequential patterns from their database. One of the most well-known methods sequential pattern mining has is a wide range of applications, including web-log analysis, customer purchase behavior analysis, and medical record analysis [1]. Customers' transaction records can be mined for sequential trends in the retail industry. For example, after purchasing a laptop, a consumer returns to purchase a mouse and a television the next time. The store may utilize this information to analyze client behavior, understand their interests, meet their expectations, and, most importantly, forecast their requirements. Sequential patterns of symptoms and diseases displayed by patients in the medical sector uncover significant symptom/disease correlations that can be a significant source of information for medical diagnosis and preventative medicine. The exploring activity of a user may be retrieved from member records or log files in Web log analysis. For example, after seeing a web page on "Data Mining," the user will return to "Business Intelligence" to look for fresh information the following time. When these sequential patterns are followed, they provide enormous benefits and enhance customer loyalty.

Retail establishments for example, frequently gather client purchase information in sequence databases, where a sequential pattern would reflect a person's purchasing behavior. Each purchase in such a database would be represented as a set of purchased goods, and a customer sequence would be a sequence of such itemsets. In more technical terms, sequential pattern mining is described as discovering all frequent subsequences that fulfill the provided minimal support threshold given a sequence database and a user-specified minimum support threshold [2].

As a vast quantity of semi-structured data is available, knowledge extraction from the World Wide Web has become a significant and demanding undertaking. Web mining is the use of data mining tools to uncover trends on the Internet. Data can be acquired at the server-side, client-side, proxy servers, or from an organization's database, which contains business data or aggregated Web data, in Web Mining. Web mining information is analyzed using typical data mining characteristics such as clustering and classification, association, and evaluation of sequential patterns. Web mining may be classified into three forms based on analytic goals: web usage mining, web content mining, and web structure mining.

Web Usage Mining has many real-world applications, such as optimizing web site designs, monitoring system performance and network communications, understanding user reaction, motivation, and constructing adaptable Websites; it is currently a highly essential and valuable subject. Web use mining is focused with discovering user navigational patterns on the World Wide Web by extracting information from web logs, where ordered sequences of events in the sequence database are formed of single things rather than sets of objects, and where a web user may physically visit only one web page at any point in time. Pattern mining and research in data mining, machine learning, and statistics are mostly focused on online pattern finding. In terms of pattern mining, it may be: Statistical analysis, which is used to collect important statistical information such as the most often viewed pages; association rule mining, which is used to locate references to a group of pages that are accessed together with a support value greater than a certain threshold; Sequential pattern mining [1] is a technique used to detect frequent sequential patterns in lists of Web sites arranged by viewing time in order to forecast visit patterns. Clustering is a technique for grouping together people that share similar traits. Classification is the process of categorizing consumers into preset groups based on their attributes.

Most web usage-mining systems now treat a user's web access as one page at a time, resulting in a particular sequence database with just one item in each sequence sorted event list. Given a collection of events E = a, b, c, d, e, f, which may represent product web pages viewed by users in an e-commerce application, a web access sequence database for four users may have four records: [SD1, acbfaec>], [SD2, efbca>], and [SD3, abfac>. A web log pattern mining on this web sequence database can find a frequent sequence, abac, indicating that more than 90 percent of users who visit product a's web page also visit product b's web page and then revisit product a's page before visiting product c's page. To encourage the sale of other items, store management may set special prices on product a's web page, which gets viewed a number of times in sequence. The web log could be server-side, client-side, or on a proxy server, each with its own set of advantages and disadvantages in terms of locating the users' relevant patterns and navigational sessions.

Related Work

Sequential Pattern Mining extracts the frequent sequential patterns from a sequence database. Apriori-based, pattern-growth, early-pruning, and hybrids of these three approaches are the several types of sequential pattern mining algorithms. Breadth-first search, generate-and-test, and numerous database scans are all significant characteristics of apriori-based approaches that pose issues and slow down algorithm performance. Pattern-growth algorithms have been extensively tested on mining the web log and found to be quick, and early-pruning methods have had success with protein sequences stored in dense databases [14]. Apriori-based methods have been shown to be too slow and have a wide search space, whereas pattern-growth algorithms have been tested widely on mining the web log and found to be quick. Traditional sequential pattern mining methods, such as Apriori-based algorithms, have the drawback of requiring repeated database searches to discover which candidates are genuinely common [1].The majority of the existing solutions for lowering the computational cost of the apriori property use a bitmap vertical representation of the access sequence database and bitwise operations to compute support at each iteration. The overheads introduced by the altered vertical databases, in turn, degrade the algorithm's performance, though not essentially to the level of pattern-growth algorithms [3].

Set Theory-based Negative Sequential Pattern mining (ST-NSP) and a related method, e-NSP, were suggested by Cao et al. [4] to effectively discover NSP by incorporating just the detected Positive Sequential Pattern (PSP) without re-scanning the database. Negative confinement is therefore described as a set theory-based method for determining if a data sequence includes a negative sequence. Secondly, an effective method for converting the negative containment problem to a positive containment problem is provided. The NSC assistance is thus computed solely on the basis of the PSP. This not only eliminates the need for extra database searches, but also allows PSP mining methods to be used to mine for NSP. Finally, a simple yet effective technique for generating NSC is presented. Theoretical studies demonstrate that e-NSP excels on datasets with a limited number of sequence elements, a large number of itemsets, and low minimum support. Through thorough trials on three synthetic and six real-life datasets, e-NSP is compared to two currently existing NSP mining techniques in terms of data features, computing costs, and scalability. e-NSP is tens to thousands of times quicker than baseline techniques, and it provides a reliable and practical method for mining NSP in huge datasets utilizing existing PSP mining methods.

Gao et al. [5] modified the limitation given in Apriori All/Some and found that a user-oriented and self-adaptive strategy outperforms a probabilistic knowledge representation. Sampling/compression, Candidate Sequence trimming, search space partitioning, tree projections, Depth first traversal, suffix/prefix growth, and memory only are some of the essential aspects of pattern growth based sequential pattern mining algorithms. When mined with low minimum support values and maximum confidence among the patterns formed, the traditional set-based apriori-based technique described by Reshamwala and Mahajan is employed since the approach has acceptable performance characteristics such as low CPU execution time and low memory use. With the implementation of the Hash Map data structure in Java, the technique executes a Breadth-first search, avoiding support counting and resulting in fewer database scans. It also aids in projecting the database in a vertical arrangement and coding the locations of the itemsets. The application of Set operations to the method leads in database shrinkage. Candidate sequence pruning is handled by the algorithm using the intersection operation, which allows them to prune candidate sequences early in the mining process. The correlations between the patterns formed are greatly increased as the database shrinks.

Pei et al. [6] created the Online Access Pattern tree (or WAP-tree), a compressed data format that makes developing algorithms for mining access patterns from web logs easier. Many changes have been proposed since then, such as the Position Coded Pre-Order Linked Web Access Pattern mining algorithm, Conditional Sequence mining method, and modified Web Access Pattern (mWAP) algorithm, to further enhance efficiency by eliminating the need for any re-construction of intermediate WAP-trees during mining.

Chiu et al. [7] developed the DISCall method in conjunction with the Direct Sequence Comparison DISC approach to eliminate counting support by trimming non-frequent sequences based on other sequences of the same length. There is yet no DISC-all variant for web log mining. Breadth-first search, generate-and-test, and numerous database scans, as explained below, are all significant elements of apriori-based approaches that pose hard difficulties and impede algorithm performance.

Chen et al. [8] suggested a time interval sequential mining technique. They introduced two efficient time-interval sequential sequence mining techniques. The first algorithm is based on the traditional Apriori technique, and the second on the PrefixSpan method. The second approach outperforms the first approach in terms of computation time and scalability when various factors are taken into account. Villafane et al. in [13] expanded sequential sequence approaches by incorporating the connection. The time between occurrences is taken into account in time interval sequential mining.

Ozden et al. [9] improved the transactional database by adding a time feature that defines when a transaction appeared and analyzed the periodic nature of the patterns to find cyclic association rules. In this work, a database is time-fragmented into non-overlapping sections. Cyclic association rules are those that appear in at least a specified number of subsets. The design of the mining method is substantially simplified by fragmenting the data and counting the number of subsets in which a pattern appears. The disadvantage is that patterns (or association rules) spanning numerous windows cannot be detected.

Using temporal data, Lee et al. [10] presented mining temporal interval relational principles. This method combines an event generalization algorithm with a time interval relation rule discovery process. The event generalization algorithm aggregates time interval events with time points and generalizes them into time interval

data. The time interval relation rule discovery method discovers frequent time interval relations using time interval data provided by the event generalization method to build time interval relation rules.
The Sequential Event Pattern Mining algorithm (SEPM) was proposed by Kanaan and Kheddouci [11]. The goal is to mine clickstream data in order to identify and evaluate useful sequential click patterns. In order to make these patterns more evident, the time spent on each page is also considered. SEPM keeps track of the durations of the objects during the mining process and extracts patterns based on the average durations of these items without requiring repeat scans of the dataset. SEPM is efficient and scalable, according to their experimental results on both real and synthetic datasets. Talpur [14] uses innovative data mining techniques to derive significant insights into visitor behavior and has integrated mobile social media data for effective collecting of tourist activity information.

Methodology
The method of identifying all subsequences that appear often in a particular dataset is known as sequential pattern mining. A sequence database consists of ordered elements or events

Table 1: Sequence

Frequent 3-Sequences
<{1}{2}{3}>
<{1}{2 5}>
<{1}{5}{3}>
<{2}{3}{4}>
<{2 5}{3}>
<{3}{4}{5}>
<{5}{3 4}>

The 3-sequences have three items. Two items can be joined if subsequences obtained by removal of first element of y1 and last element of y2 are same. Therefore, the possibilities of the sequences in Table 1 are as follows:

Sequence	**Subsequences**
<{1}{2}{3}>	<{1}{2}{3}>
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 1: Possibilities of Sequence {1}{2}{3}

Sequence	**Subsequences**
<{1}{2 5}>	<{1}{2}{3}>
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 2: Possibilities of Sequence {1}{2 5}

Sequence	**Subsequences**
<{1}{5}{3}>	<{1}{2}{3}>
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 3: Possibilities of Sequence {1}{5}{3}

Sequence	**Subsequences**
<{2}{3}{4}>	<{1}{2}{3}>
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 4: Possibilities of Sequence {2}{3}{4}

Sequence	Subsequences
<{2 5}{3}><{1}{2}{3}>	
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 5: Possibilities of Sequence {2 5}{3}

Sequence	Subsequences
<{3}{4}{5}><{1}{2}{3}>	
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 6: Possibilities of Sequence {3}{4}{5}

Sequence	Subsequences
<{5}{3 4}><{1}{2}{3}>	
<{1}{2 5}>	
<{1}{5}{3}>	
<{2}{3}{4}>	
<{2 5}{3}>	
<{3}{4}{5}>	
<{5}{3 4}>	

Figure 7: Possibilities of Sequence {5}{3 4}

To obtain the candidate generation, remove first item from sequence and last item from the subsequences, then identify the subsequences that have same item with sequence and join them.

Sequence	Subsequences
<{2}{3}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 8: Sequence {2}{3}

Sequence	Subsequences
<{2 5}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 9: Sequence {2 5}

Sequence	Subsequences
<{5}{3}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 10: Sequence {5}{3}

Sequence	Subsequences
<{3}{4}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 11: Sequence {3}{4}

Sequence	Subsequences
<{5}{3}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 12: Sequence {5}{3}

Sequence	Subsequences
<{4}{5}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 13: No Match Sequence

Sequence	Subsequences
<{3 4}><{1}{2}>	
<{1}{2}>	
<{1}{5}>	
<{2}{3}>	
<{2 5}>	
<{3}{4}>	
<{5}{3}>	

Figure 14: No Match Sequence

Candidate generation illustrated in Table 2 is obtained by joining items in subsequences that have same value with items in sequence.

Table 2: Candidate Generation

Candidate Generation
<{1}{2}{3}{4}>
<{1}{2 5}{3}>
<{1}{5}{3 4}>
<{2}{3}{4}{5}>
<{2 5}{3 4}>

Prune candidates that contain a subsequence which is infrequent in K-1 subsequences. To achieve this; produce sequences for all generated candidate in Table 2. Generate subsequences by removing one item from each sequence.

Candidate	Sequences	Subsequences
<{1}{2}{3}{4}>	<{1}{2}{3}{4}>	<{2}{3}{4}>
<{1}{2}{3}{4}>		<{1}{3}{4}>
<{1}{2}{3}{4}>		<{1}{2}{4}>
<{1}{2}{3}{4}>		<{1}{2}{3}>

Figure 15: Candidate {1}{2}{3}{4}

Candidate	Sequences	Subsequences
<{1}{2 5}{3}> <{1}{2 5}{3}> <{1}{2 5}{3}> <{1}{2 5}{3}>	<{1}{2 5}{3}>	<{2 5}{3}> <{1}{5}{3}> <{1}{2}{3}> <{1}{2 5}>

Figure 16: Candidate {1}{2 5}{3}

Candidate	Sequences	Subsequences
<{1}{5}{3 4}> <{1}{5}{3 4}> <{1}{5}{3 4}> <{1}{5}{3 4}>	<{1}{5}{3 4}>	<{5}{3 4}> <{1}{3 4}> <{1}{5}{4}> <{1}{5}{3}>

Figure 17: Candidate {1}{5}{3 4}

Candidate	Sequences	Subsequences
<{2}{3}{4}{5}> <{2}{3}{4}{5}> <{2}{3}{4}{5}> <{2}{3}{4}{5}>	<{2}{3}{4}{5}>	<{3}{4}{5}> <{2}{4}{5}> <{2}{3}{5}> <{2}{3}{4}>

Figure 18: Candidate {2}{3}{4}{5}

Candidate	Sequences	Subsequences
<{2 5}{3 4}> <{2 5}{3 4}> <{2 5}{3 4}> <{2 5}{3 4}>	<{2 5}{3 4}>	<{5}{3 4}> <{2}{3 4}> <{2 5}{4}> <{2 5}{3}>

Figure 19: Candidate {2 5}{3 4}

Compare the subsequences that are available in frequent 3-sequence table. However, in these possibilities there is only one subsequence which has all the support available in frequent 3-sequences. Thus the candidate pruning is <{1}{2 5}{3}>

Data

The data set contains all the transactions occurring between 01/12/2010 and 09/12/2011 for a UK/Ireland based and registered non-store online retail. The company mainly sells unique all-occasion gifts. Many customers of the company are wholesalers. The dataset undergoes exploratory data analysis and preprocessing.

Results

Analysis was carried out on UK and Ireland dataset to determine the most frequent purchased product in UK and Ireland. This analysis requires that all the data for a transaction sequence be included in 1 row (like a receipt), which includes one receipt number and also the groceries list. The result is sorted in descending order, means the strength of relationship is sorted from the strongest and weaker.

Table 3: Analysis of UK Data Set

Invoice	Antecedents	Consequents	Antecedent support	Consequent support	Support
0	Roses regency teacup and saucer, green regency teacup	Pink regency teacup and saucer	0.030957	0.031803	0.022177
2	Roses regency teacup and saucer, pink regency teacup	green regency teacup and saucer	0.024909	0.039802	0.022177
4	pink regency teacup and saucer	green regency teacup and saucer	0.031803	0.039802	0.026275
8	green regency teacup and saucer, pink regency teacup	Roses regency teacup and saucer	0.026275	0.043900	0.022177
11	pink regency teacup and saucer	Roses regency teacup and saucer	0.031803	0.043900	0.024909
12	green regency teacup and saucer	Roses regency teacup and saucer	0.039802	0.043900	0.030957
13	Roses regency teacup and saucer	green regency teacup and saucer	0.043900	0.039802	0.030957
14	Gardeners kneeling pad cup of tea	Gardeners kneeling pad keep calm	0.040713	0.048192	0.029787

Support of (ROSES REGENCY TEACUP AND SAUCER, GREEN REGENCY TEACUP AND SAUCER) and PINK REGENCY TEACUP AND SAUCER is 0.022177, it means they appear together in 2.2177% of transactions. In other words, on 10000 transactions there are on average 221.77 transactions with both together. We can make the hypothesis that it’s easier to sell ‘ROSES REGENCY TEACUP AND SAUCER’ to someone buying ‘GREEN REGENCY TEACUP AND SAUCER’.

For Ireland data set, the setting of minimum support changed since the number of observations is smaller rather than the UK data set.

Table 4: Analysis of Ireland Data Set

Invoice	Antecedents	Consequents	Antecedent support	Consequent support	support
93797	Regency cakestand 3 tier, pink regency teacup	Roses regency teacup and saucer, green regency teacup	0.081967	0.127049	0.077869
97460	Regency cakestand 3 tier, pink regency teacup	Green regency teacup and saucer	0.081967	0.139344	0.077869
97461	Regency cakestand 3 tier, roses regency teacup	Green regency teacup and saucer	0.081967	0.139344	0.077869
102900	Regency cakestand 3 tier, green regency teacup	Roses regency teacup and saucer	0.077869	0.180328	0.077869
102936	Regency cakestand 3 tier, green regency teacup	Roses regency teacup and saucer	0.094262	0.180328	0.094262
102939	Regency cakestand 3	Roses regency	0.081967	0.180328	0.081967

	tier, green regency teacup	teacup and saucer			
104176	Pink regency teacup and saucer	Roses regency teacup and saucer	0.114754	0.180328	0.110656
104185	Green regency teacup and saucer, pink regency teacup	Roses regency teacup and saucer	0.106557	0.810328	0.102459

The analysis shows that both ROSES REGENCY, REGENCY CAKESTAND 3 TIER and GREEN REGENCY have support = 0.77869. The marketing strategy should combine both items in product bundling and discount promotion.

Conclusion

We conducted our investigation using actual online retail transaction data from the United Kingdom and Ireland. The findings of this market basket study might be utilized to improve retail strategy. There are marketing insights that may be optimized; here is the Marketing Mix concept for things with a strong relationship: We can put the goods with strong relationships next to each other on the homepage, we can place them together with extra discounts or develop a new promotion plan, and we can compute the price strategy for the products.

We propose that UK wholesalers pair these teacups and saucers together: ROSES REGENCY, GREEN REGENCY, and PINK REGENCY. Another suggestion is that they provide a discount on ROSES REGENCY or GREEN REGENCY when customers purchase PINK REGENCY. Best of all, clients would be delighted to pay a lesser price for the product bundle consisting of these teacups and saucers.

According to the study, ROSES REGENCY, REGENCY CAKESTAND 3 TIER, and GREEN REGENCY have support = 0.77869 for Ireland wholesalers. Product bundling and price promotion should be included in the marketing plan.

References

[1]. R. Agrawal and R. Srikant. “Mining sequential patterns,” In 11th Int’l Conf. of Data Engineering (ICDE’95), pp. 3-14, Taipei, Taiwan, Mar. 1995.

[2]. R. Agrawal and R. Srikant. Fast algorithms for mining association rules in large databases. In Proceedings of the 20th International Conference on Very Large Databases. Santiago, Chile, pp. 487499, 1994.

[3]. S. Song, H. Hu and S. Jin. HVSM: A new sequential pattern mining algorithm using bitmap representation. In Advanced Data Mining and Applications. Lecture Notes in Computer Science, vol. 3584, Springer, Berlin, 455–463. 2005.

[4]. L. Cao, X. Dong and Z. Zheng. Efficient negative sequential pattern mining. Elsevier B.V. www.elsevier.com/locate/artint, pp.156–182, 2016.

[5]. S. Gao, R. Alhaji, J. Rokne and J. Guan. “Set Based Approach in Mining Sequential Patterns”, 24th International Symposium on Computer and Information Sciences, 2009. ISCIS 2009, pp 218 – 223.

[6]. J. Pei, J. Han, B. Mortazavi-Asl, and H. Zhu. Mining access patterns efficiently from web logs. In Proceedings of the Paci_c-Asia Conference on Knowledge Discovery and Data Mining (PAKDD00). Kyoto, Japan, pp. 396-399, 400-402, 2000.

[7]. D. Y. Chiu, Y. H. Wu and A. L. P. Chen. An efficient algorithm for mining frequent sequences by a new strategy without support counting. In Proceedings of the 20th International Conference on Data Engineering. 375–386. 2004.

[8]. Y. L. Chen, M. C. Chiang and M. T. Ko. Discovering time-interval sequential patterns in sequence databases. Expert System Application, 25(3), 343-354, 2003.

[9]. B. Ozden, S. Ramaswamy and A. Silberschatz. Cyclic association rules. In ICDE, 412–421, 1998.

[10]. Y. J. Lee, J. W. Lee, D. J. Chai, B. H. Hwang and K. H. Ryu. Mining temporal interval relational rules from temporal data. The Journal of Systems and Software, 82, 155–167, 2009.

[11]. M. Kanaan and H. Kheddouci. "Mining Patterns with Durations from E-commerce Dataset". Complex Network, Dec 2018, Cambridge, United Kingdom.
[12]. A. Reshamwala and S. Mahajan. “Traditional Set based Approach in Mining Sequential Patterns”, Proceedings of National Conference on New Horizons in IT- NCNHIT 2013, ISBN 978-93-82338, pp. 173- 177.
[13]. R. Villafane, K. A. Hua, D. Tran and B. Maulik. Knowledge discovery from series of interval events. Intelligent Information Systems, 15(1), 71–89, 2000.
[14]. A. Talpur. "Mining Tourist Behavior: A study of Tourist Sequential Activity Pattern through Location Based Social Networks". VU RESEARCH Institute for Sustainable Industries & Liveable Cities, Victoria University, 2019.