A Novel Frequent Pattern Mining Algorithm for Evaluating Applicability of a Mobile Learning Framework

H. K. Salinda Premadasa

Received: 1 January 1970 Accepted: 1 January 1970 Published: 1 January 1970

Abstract

The applicability of a mobile learning system reflects how it works in an actual situation under diverse conditions. In previous studies, researches for evaluating applicability in learning systems using data mining approaches are challenging to find. The main objective of this study is to evaluate the applicability of the proposed mobile learning framework. This framework consists of seven independent variables and their influencing factors. Initially, 1000 students and teachers were allowed to use the mobile learning system developed based on the proposed mobile learning framework. The authors implemented the system using Moodle mobile learning environment and used its transaction log file for evaluation. Transactional records that were generated due to various user activities with the facilities integrated into the system were extracted. These activities were classified under eight different features, i.e., chat, forum, quiz, assignment, book, video, game, and app usage in thousand transactional rows. A novel pattern mining algorithm, namely Binary Total for Pattern Mining (BTPM), was developed using the above transactional dataset’s binary incidence matrix format to test the system applicability. Similarly, Apriori frequent itemsets mining and Frequent Pattern (FP) Growth mining algorithms were applied to the same dataset to predict system applicability.

Index terms — system applicability, mobile learning, frequent pattern mining, apriori algorithm, fp growth algorithm

I. Introduction

Mobile learning (ML) emerged as an essential and successful learning method due to its flexibility of time and place for learning in the present globe. On the other hand, learners and teachers have an interactive and cost-effective learning environment for carrying out educational activities with cutting-edge technology advancements. However, at the same time, learners and teachers experience various limitations in this learning method, such as the quality of the learning applications. Therefore, it is vital to evaluate ML applications in various dimensions, such as their ML applicability [1]. The Cambridge Oxford Dictionary defines applicability as "the fact of affecting or relating to a person or thing." Applicability definitions for domains related to computer systems are minimal. Hence we found a specific definition for the public health sector domain, and it defines applicability as "the extent to which an intervention process could be implemented in another" [2]. On the other hand, transferability gives a meaning similar to applicability. It defines transferability as "the extent to which the measured effectiveness of an applicable intervention could be achieved in another setting." Rosemann and Vessey (2008) proposed the significance and capability to check the applicability of a research framework with three factors, i.e., importance, accessibility, and suitability [3]. Accordingly, we can recognize an applicable mobile learning system (MLS) as "the extent to which a MLS could be implemented effectively in a target environment as well as in another environment with a similar setting." This study’s main objective is to evaluate the applicability of the proposed mobile learning framework (MLF). This framework consists of seven independent variables, and each variable has different influencing factors. Evaluating this MLF for applicability would help realize the proposed framework’s applicability when it works in a real-world environment. Normally, pattern mining algorithms are used to describe
patterns in usage behaviors [4]. Hence, we use pattern mining to study users’ usage of different learning tools in the system, such as chat, forum, quizzes, assignments, etc. Proper understanding of these usages would help calculate how many users apply these tools in their academic activities in the proposed system. Considering each tool’s usage would support predicting the system’s overall applicability (Fig. 3: motivating example). Because if a learning system has better applicability, its users should fully practice the tools integrated into it. In this study, the authors intend to analyze the transactional log of the MLS, developed using the proposed MLF for evaluating applicability. According to previous studies, the FP-Growth [5] efficient pattern mining and Apriori [6] popular association rule mining algorithms have been utilized to evaluate many automated systems. FP-Growth algorithm was used to evaluate systems to identify important segments in learning materials [7], learning behaviors [8], best course modules [9], learners’ requirements [10], offensive activities, and illegal transactions [11]. In these studies, the FP-Growth algorithm helps to improve the learning process, course selection [9], course recommendation [10], security enhancement, and legal judgments [11] in learning systems and portals. On the other hand, the Apriori algorithm was used to evaluate systems to identify, information for decision making [12], tourists attractive places [13], course administration history [9], study preferences, archived cyber-attacks, and network hackings [14], requirements in software [15], health clinical information in clinical [16], and deformation states of landslides [17]. In these studies, the Apriori algorithm helped in different activities such as better administrative decisions [12], improving tourist attraction [13], enhancing course facilities, offering productive subjects [9], preventing networks attacks [14], recommend software requirements [15], enhanced health decision and treatments [16], and predict landslides accurately [17]. But it is a very shortage of using FP Growth or Apriori algorithms in the evaluation applicability of learning systems. Also, difficult to find such applicability evaluation using any statistical approach too in previous studies. Moreover, these two well-established algorithms have no direct provisions for evaluating the applicability of the system. Nevertheless, using the rules and frequent patterns generated by Apriori and FP Growth algorithms can predict applicability. In this study, the authors proposed a novel frequent pattern mining algorithm for applicability evaluation directly in the proposed learning system to address this research gap. Also, FP Growth and Apriori algorithm are used to predict the applicability of the same system.

2 II. Related Work

Pattern mining is a popular data mining tactic to emerge secret knowledge in data stores, and it is applied for solving issues arising in various scenarios. A Pattern mining approach was proposed to identify the most critical or complex learning segments in video tutorials. The proposed method integrates a learning model that learns the above-considered components of the analyzed video transaction log. A pattern mining framework was proposed to identify learners’ different learning behaviors and improve their learning process with the institutional learning system. This study’s output reveals that the learners obtained significant study performance in different learning modes via various tools integrated into the system [8]. Another pattern mining framework was proposed to analyze huge databases by addressing existing pattern mining algorithms such as a large number of searching iterations, excessive space for processing, and too much time requirements. Results reveal that this approach can solve different kinds of pattern mining problems [18]. An online course recommender system was proposed using the FP-Growth algorithm to guide learners to select learning courses according to their preferences. The investigation displays that the system has better efficiency in instructing learners for selecting appropriate learning materials in their learning process [9]. FP-Growth-based data mining technique was used to promote educational services in educational institutes. Various variables related to learners and educational institutes were employed on FP-Tree to determine regular data items. Research reveals that the best selection of attributes in the data for the algorithm gives better results [19]. FP-Growth algorithm was used to elaborate users’ access patterns in learning portals. The study revealed, such as learners’ favorite courses, less and high navigation areas of the learning site, and recommendations for advancing both learner’s gain and user-friendliness of the site [10]. Wu and Zhang (2019) researched to extract support information to prove offensive actions. An improved FP-Growth algorithm analyzed data associated with illegal transactions and supported legal judgment with better efficiency and accuracy [11].

Another method for hidden information recovery from large databases is associate rule mining, and its applications are spread in various researches in the past. The E-learning system was evaluated by combining the association rule mining method and the fuzzy analytic hierarchy process [20]. Apriori algorithm-based association rule mining was used to analyse the Wi-Fi data in attractive tourist places. The study results give the association rules of travel patterns of tourists’ movements and enable further enhancements of tourist’s magnetism of travel destinations [13]. Learner assisting study guides approval mechanism was proposed using association rule mining with archived course administration data. This approach queries useful relationships in learners’ learning subject preferences [9]. An improved Apriori algorithm is offered to find fresh network attacks using the data that was produced by previous episodes. This method has optimal accuracy with superior efficiency for discovering cyber strikes [14]. A customized Apriori algorithm was used to improve the audit system’s security building association rules using fewer scanning cycles in logs with shorter processing time [21]. Apriori algorithm-based recommender system was proposed to enhance the accuracy of requirements in software development [15]. Apriori algorithm was customized to excavate intelligence information from virtual reality applications for effective decision making [22]. Apriori algorithm was executed successfully in analyzing the deformation states of landslides [17]. A heart disease prediction model was proposed by applying the Apriori algorithm in clinical datasets [23]. Here, we
used the Apriori associate rule mining and FP-Growth frequent pattern mining algorithm-based approaches as comparable methods to the novel proposed pattern mining algorithm for evaluation applicability of the proposed MLS.

3 III. Preliminaries
In this section, the authors discuss the proposed MLF and its implemented MLS, which will be evaluated for applicability. Also, the critical theories undergo related to the proposed solution for evaluating the system’s applicability, such as associate rule mining, Apriori, and FP-Growth algorithms.

4 a) Mobile Learning Framework for Higher Education
We can refer to many frameworks that were implemented in the Moodle environment successfully in previous studies. Halvoník and Kapusta (2020) implemented an e-Learning material composing framework through Moodle LMS with the teachers’ highest inclination [24]. A framework for evaluating student learning was performed through the Moodle platform to create useful tests for learners [25]. Karagiannis and Satratzemi (2017) proposed a Moodle implemented framework for e-learning content adaptation according to learner wishes and better usability with decent learning outcomes [26]. Hence, according to the literature, several conceptual frameworks have been implemented via the Moodle mobile learning environment (MMLE). So in our study, we implemented the proposed conceptual MLFrame through the MMLE.

In this study, we require to measure the applicability of the MLS implemented based on the proposed MLF for higher education (MLFrame). This framework consists of seven independent variables: learner, teacher, mobile ML devices, ML tools, ML contents, higher education institutes, and communication technology (Fig. ??). Each independent variable has several influencing factors: motivation, usefulness, interactivity, ease of use, etc. These factors were realized through various resources and facilities embedded in the MLS such as chat, forum, games, quizzes, assignments, etc. [27], [28]. This MLFrame was implemented in the MMLE by integrating new facilities and modifying the existing Moodle mobile application. Therefore, existing Moodle plugins were enhanced to develop these new ML facilities for the Moodle mobile environment [28]. Because most of the facilities available in the Moodle learning environment are not implemented in the Moodle mobile environment. Hence, when implementing the MLFrame in the Moodle mobile environment, we customized the Moodle ML application by upgrading relevant plugins to serve the facilities introduced in the MLFrame. ?? depicts the concept associated with predicting the system applicability. Consider a MLS with facilities chatting, forum, quiz, assignment, book, video, game, and app_usage. If this app was allowed to use 100 particular users, then assume transaction log analysis as follows. 8,7,6,5 features out of all 8 features used by 60,10,15,5 users, respectively. Therefore, 70 users out of 100 users used at least 7 features out of all 8 features (or 88% features). Hence system applicability was 70% for 88% feature usage. Further, 85 users out of 100 users used at least 6 features out of 8 features (or 75% features). Hence system applicability is 85% for 75% features usage. Similarly, it can be realized that the system applicability is 90% for 63% features usage.

5 Fig. 3: Example for Predicting Applicability using Pattern Mining Algorithm c) Associate Rule Mining
Association rule mining is used to find essential associations among stored data in large databases. ‘Antecedent’ and ‘Consequent’ are significant two fractions in association rules. These are data items finding in databases, and ‘Antecedent’ combines with the ‘Consequent’. Moreover, finding frequent itemsets is a crucial requirement to mine association rules from databases. Additionally, two important factors such as support and confidence, must be defined when finding association rules using these frequent itemsets [29].

6 d) Apriori Algorithm
Apriori is a Latin term which denotes "from what comes before". Bottom-up and breadth-first search strategies are taken into account. Agarwal and Srikant (1994) developed the Apriori algorithm for generating associate rules by frequent pattern mining. The main terminologies used in the Apriori algorithm are Min_supp, Min_conf, Frequent itemsets, Apriori Property, Join Operation, Join Step, and Prune Step [30].

7 e) Frequent Pattern (FP) Growth Algorithm
FP-Growth is a widespread pattern mining algorithm used in data mining. In this algorithm, frequent patterns are stored in a tree-like data structure called FP-tree. The algorithm’s main steps are calculating each database item’s support count by scanning, deleting irregular patterns, and order remains. Then FP-tree is constructed and frequent patterns are generated using FP-tree [5]. This improves the frequent pattern mining technique because it scans the database only twice. There is no candidate set generation, though it is not suitable for mining patterns in databases that are updated frequently [31].
9 B) ALGORITHM IMPLEMENTATION WITH A SAMPLE

8 a) Process of the Proposed Algorithm
The processing steps of the proposed algorithm implementation are described below.

Procedure 1: Purpose is generating a features list (extracting unique features, used in this evaluation, stored in the transaction log of the system)
Step 01 & 09: For loop iterates from 1st row to last row of the transactional database.
Step 02 & 08: For loop iterates from 1st item to the last item of a transactional database row.
Step 03 & 07: For loop iterates from 1st row to the last item of the array call Array Features. Array Features is an array that contains unique features of the transactional database.
Step 04: Check whether each item does not exist in the array Array Features (f m).
Step 05: If the new item is not in the array, the item is saved in Array Features as the last item (f s+1).
Step 14: Check each item in the transaction database (a ij) with feature items in Array Features (f m).
Step 15: If a similar item is found, replace the value in the Array Dataset BIMF with 1. The position of the item (i.e. d im) substituting in the Array Dataset BIMF is found by taking the associate item’s row number of the transactional database (i th row) and column number of the Array Features (m th column).
Likewise, replace 0 with 1 in Array Dataset BIMF to denote similar features in the same column in Array Dataset BIMF for all items in the transactional database. The second last element (n+1) of each row of the array is the Total Decimal Value of binary digit positions in entire Transactions (tdvt i). tdvt i = ? ?? ???? ????-1

\[ ????? ??=0 \]
\[ 2 ?? ; \text{where} \ d \ ij \text{is the digit value of} \ j \text{th position in} \ i \text{th transaction i.e.,} \ d \ ij =\{0,1\} \]
The last element (n+2) of each row of the array is the Count of the Non-zero Binary digits in each transaction row (cnb i). cnb i = ? ?? ???? ????-1 ??=0

\[ ; \text{where} \ d \ ij \text{is the digit value of} \ j \text{th position in} \ i \text{th transaction i.e.,} \ d \ ij =\{0,1\} \]
Step 26: At the end of each row, check whether the percentage of features used in a transaction is greater than or equal to minFUT. The minFUT is the given threshold value for the percentage of minimum features usage in a transaction. For instance, if minFUT = 70% for a 10 features transaction means, at least 7 features are used out of 10 features by a user.

\[ (\text{If} \ j = n \text{and} \ cnb i *(100/n) >= \text{minFUT then}) \]
If the above condition is satisfied, then save the transaction pattern to another array (i.e., ArrayPatterns).
In each row of ArrayPatterns[tdvt k, pattern Count k], tdvt k denotes total decimal values of binary digit positions in entire transactions. Feature usage transaction patterns can be derived by taking the binary conversion of the tdvt i. And patternCount k gives the count of the number of occurrences of the pattern.
Step 27: For loop iterates from 1st row to last row of the ArrayPatterns[]. Step 39: Percentage of each distinct usage transaction pattern can be calculated by using the equation, Percentage of k th pattern = (patternCount k / t) * 100

Step 41: Finally, the overall percentage of transactions whose FUT is greater than or equal to minFUT can be calculated by taking the percentage of summation of the element ArrayPatterns[(patternCount k)]. i.e., \[ \sum_{k=1}^{n} \text{patternCount} k \]

Step 06: The table 'ArrayTDVT' is created by adding two columns at the end to the 'ArrayDatasetBIMF'.

\[ \text{The last element} \ (n+2) \text{of each row of the array is the Count of the Non-zero Binary digits in each transaction row} \]
\[ \text{(If} \ j = n \text{and} \ cnb i *(100/n) >= \text{minFUT then}) \]
In each row of ArrayPatterns[tdvt k, pattern Count k], tdvt k denotes total decimal values of binary digit positions in entire transactions. Feature usage transaction patterns can be derived by taking the binary conversion of the tdvt i.

\[ \text{And patternCount} k \text{gives the count of the number of occurrences of the pattern.} \]

\[ \text{Step 27: For loop iterates from 1st row to last row of the ArrayPatterns[]} \]
Step 39: Percentage of each distinct usage transaction pattern can be calculated by using the equation, Percentage of k th pattern = (patternCount k / t) * 100

Step 41: Finally, the overall percentage of transactions whose FUT is greater than or equal to minFUT can be calculated by taking the percentage of summation of the element ArrayPatterns[(patternCount k)]. i.e., \[ \sum_{k=1}^{n} \text{patternCount} k \]

9 b) Algorithm Implementation with a Sample
Fig. 4 depicts the generation of the required dataset by implementing the proposed algorithm to evaluate the applicability of MLFrame. In this example, five features were considered for the transaction database. Moodle tools usage and users tables of the log database of the system were used to create the transaction database.
Step 01: 'Transaction' table is created by copying user ids from the main user table and updating each user id from the tools usage log table in the database of the MLS.
Step 02: 'ArrayFeatures' table is created by inserting distinct features in the 'Transaction' table. As we use only 5 features, this ArrayFeature table consists of a row with 5 values.
Step 03: 'ArrayDatasetBIMF' consists of 5 rows (transactions of 5 users) and 6 columns (the first column is for TID-transaction ids and the rest of the rows for features). The array consists of user ids and 5 tools (features) same order with the 'ArrayFeatures' in a particular row.
Step 04: The table 'ArrayTDVT' is created by adding two columns at the end to the 'ArrayDatasetBIMF'.
New columns are 'TDVT,' and 'CNB' represents the decimal value of the binary values in the same row and the count of non-zero binary digits in the same column, respectively.
Step 05: Different feature usage patterns with frequency are stored in the array called 'ArrayPatterns'. Unique feature usage patterns are selected by satisfying the condition, each pattern’s minimum number of features are greater than or equal to the minimum feature usage percentage (minFUT) supplied. Existing Pattern Mining Algorithms Popular algorithms for pattern mining such as Apriori and FP-Growth generate itemsets or candidate itemsets to find frequent itemsets. These itemsets are a group of transactional items that reside in the transaction database. Then, they use a minimum threshold value to minimize or prune itemsets to reduce data considered in mining. On the contrary, the BTPM algorithm uses an entirely mathematical technique, i.e., binary incidence
The proposed MLS was given to precisely 1000 users, who are learners and teachers in the University of Kelaniya’s four faculties. Among them, 220 students from each faculty of Science and Commerce & Management, and 200 students from each Faculty of Social Sciences and Humanities. Also, 160 teachers participated, and they consisted of 40 teachers in each faculty mentioned above. They were asked to use the system for around 50 days.

This study was conducted according to the research framework illustrated in Fig. 5. The standard log file was extracted, and approximately half a million records were identified as different transactions related to the above user group on the given spell. Transactions were categorized into eight transactions with facilities integrated into the proposed ML application, i.e., chat, forum, quiz, assignment, book, video, game, and app_usage. The app_usage represents general user activities associated with the mobile application, such as page viewing, information modifying and deleting, etc. These activities were classified according to each user (Table 6). Finally, the transactional dataset was generated using the above eight features for all 1000 users. This dataset completed preprocessing steps to be perfect for applying algorithms such as filling in missing data and removing unusual data. This dataset consists of 1000 records, and each record represents different transactions done using the proposed ML app.

The proposed data mining-based novel frequently pattern mining algorithm was primarily implemented using Python programming language to describe the proposed MLS’s applicability (Table 1, BTPM Algorithm), and it was applied to the dataset. This algorithm caters to finding patterns of feature usage in transactions and calculating percentages of each different transaction pattern. For instance, what is the percentage of different patterns including all the features considered above (i.e., chat, forum, quiz, assignment, book, video, game, and app_usage), or what are the percentages of different patterns including seven features out of the eight features considered above? Then overall applicability of the system can be predicted by considering these transaction patterns and taking the summation of their percentage values. Next, the Apriori algorithm for associate rule mining and the FP-Growth algorithm for frequent pattern mining were applied on the same dataset and generated the best possible rules which can describe the overall systems’ applicability.

Finally, the proposed BTPM algorithm’s performance was compared with Apriori and FP-Growth algorithms. For that, a data set with 15000 records were used. The dataset was created by using the original data set multiple times with changing the order of records randomly. Each algorithm’s execution times were recorded by changing number of execution records (i.e., 2500, 5000, 7500, 10000, 12500, 15000) and percentage of minimum feature usage (supports) (20%, 40%, 60%, 80%).

Various activities in the MLS are used as eight features in the transactional dataset. Feature descriptions are mentioned in Table 2.

In this study, we use a data set including 10000 transactional rows of 1000 users. Each transaction row consists of 8 features denoting activity usage in the MLS. The second subprocedure in the proposed algorithm converts the transactional dataset to the BIMF dataset, and part of the BIMF dataset is shown below (Table ??).

The proposed data mining-based novel frequently pattern mining algorithm was primarily implemented using Python programming language to describe the proposed MLS’s applicability (Table 1, BTPM Algorithm), and it was applied to the dataset. This algorithm caters to finding patterns of feature usage in transactions and calculating percentages of each different transaction pattern. For instance, what is the percentage of different patterns including all the features considered above (i.e., chat, forum, quiz, assignment, book, video, game, and app_usage), or what are the percentages of different patterns including seven features out of the eight features considered above? Then overall applicability of the system can be predicted by considering these transaction patterns and taking the summation of their percentage values. Next, the Apriori algorithm for associate rule mining and the FP-Growth algorithm for frequent pattern mining were applied on the same dataset and generated the best possible rules which can describe the overall systems’ applicability.

Finally, the proposed BTPM algorithm’s performance was compared with Apriori and FP-Growth algorithms. For that, a data set with 15000 records were used. The dataset was created by using the original data set multiple times with changing the order of records randomly. Each algorithm’s execution times were recorded by changing number of execution records (i.e., 2500, 5000, 7500, 10000, 12500, 15000) and percentage of minimum feature usage (supports) (20%, 40%, 60%, 80%).

Various activities in the MLS are used as eight features in the transactional dataset. Feature descriptions are mentioned in Table 2.

In this study, we use a data set including 10000 transactional rows of 1000 users. Each transaction row consists of 8 features denoting activity usage in the MLS. The second subprocedure in the proposed algorithm converts the transactional dataset to the BIMF dataset, and part of the BIMF dataset is shown below (Table ??).
7 features in transactions is 51%, and the maximum percentage for six features usage in transactions is 61%.
Finally, using the Apriori algorithm results, we can assume that the proposed system’s overall applicability should be greater than 61% when at least 6 features are used in a transaction.

**14 d) Results of the FP-Growth algorithm**

FP-Growth algorithm gives the following patterns as frequent patterns for the above dataset with 40% as both minimum threshold values for support and confidence parameters. These results denote that 40% of users use at least seven features among the eight features. Also, 60% of users use at least six features among the eight features. These results secure the 75% usage of the features by 60% of users. Therefore, we can assume that the applicability of the system is not less than 60% when minimum feature usage in a transaction is equal to 75% for the FP-Growth algorithm.

**15 e) Evaluating the Algorithm**

The performance of the proposed algorithm for pattern mining was compared with Apriori and FP-Growth algorithms. The results clearly show the better efficiency of the proposed algorithm. According to the graphs in Fig. 6, the proposed algorithm (BTPM) takes less execution time than the Apriori and FP-Growth algorithm for different support thresholds in each size of transactions. Reasons for these performances in the proposed algorithm are, it scans the database only once to develop an array of features. Also, the BIMF dataset and mathematical process have quicker processing power in the proposed algorithm. On the contrary, other algorithms considered in this study use techniques to mine frequent patterns such as scans database iteratively, creating candidate itemsets, and a frequent pattern tree. However, if our proposed algorithm uses the BIMF dataset directly, these execution times reduce further.

**16 Global**

**17 f) Time complexity of the proposed algorithm**

The time complexity of an algorithm is the time estimation to execute the programming code inside the algorithm. It depends on the building blocks or control structures used in the algorithm, such as sequence, selections, and iterations. Further, the situation differs when the input is increasing. According to table 8, the total time complexity is equal to the summation of time complexity of procedure 1, procedure 2, procedure 3, and procedure 4. It is close to $O(N^2)\) . Therefore the time complexity of the proposed BTPM algorithm is realized approximately as $O(N^2)\) .

According to the previous studies, the Apriori algorithm’s time complexity was calculated as $O(N^2)\) for a larger dataset [33]. But Tahyudin and colleagues (2019) show the time complexity of the Apriori algorithm as $O(2^N)\) [34]. Also, the FP-Growth algorithm’s time complexity is $O(N \log N)\) for higher data volumes, while in lower datasets, it shows $O(N)\) better performance [35]. According to Fig. 7, the proposed BTPM algorithm’s time complexity has similar or a little better to the Apriori algorithm for more extensive data volumes. The purpose of creating the BTPM algorithm is to evaluate the proposed algorithm for applicability. BTPM algorithm provides the applicability of the MLS directly. But when Apriori and FP Growth algorithms are used to evaluate applicability, it requires certain assumptions, as mentioned in 6.3 and 6.4.

**18 VII. Conclusion and Implication**

This study’s primary purpose is to check whether the proposed MLF for higher education is applicable for higher education learners and teachers. The framework was implemented via a modified MMLE. This study was carried out using generated MySQL standard system log files integrated with a Moodle learning management system. In this study, the authors...
Figure 1: Fig. 1 :Fig. 2 :

Figure 2: 4 2023 A

Figure 3: Procedure 3 :
Figure 4: Step 28: Procedure 4: Step 38:
Figure 5: Fig. 4:
VII. CONCLUSION AND IMPLICATION

Figure 6: © 2023 Global Journals 8 2023 A

Figure 7: Fig. 5:
Array Features consists of an array of distinct features in the transactional database.

Procedure 2: Step 10: Creating an array calls Array Data Set BIMF[] for saving the transactional database in binary incidence format array size is:

t x s; t means the number of transactions in the transactional database; s means the number of features in the array Features[]. Assign 0 for all the array elements.

Step 11 & 19: For loop iterates from 1st row to last row of the transactional database

Step 12 & 18: For loop iterates from 1st item to the last item of a transactional row

Step 13 & 17: For loop iterates from 1st item to the last item of the array called Array Features[]. (Note: Array Features[] contains distinct features of the transactional database)

Figure 8:

\[
\begin{align*}
?? \times ? = 1 \\
? \times ? \times 100
\end{align*}
\]

Where \(L = \text{No. of patterns} = \text{Number of rows in the ArrayPatterns, } t = \text{no. of transactions in the dataset}\)

Figure 9: Table 1:
Feature | Type | Description
---|---|---
ID No. | Identification | Student identification number
Chat | Activity | Chat facility for discussions
Forum | Activity | Forum facility for knowledge sharing
Quiz | Activity | Quiz facility to test learners’ knowledge
Assignment | Activity | Assignment for extra academic tasks
Book | Activity | Book-facility for further reading
Video | Activity | Video facility for academic activities
Game | Activity | Game facility for academic activities
App_usage | Activities | General activities in the mobile app such as view, (View/Modify/..) modify..

Figure 10: Table 2:

<table>
<thead>
<tr>
<th>PNo</th>
<th>F1</th>
<th>F2</th>
<th>F3</th>
<th>F4</th>
<th>F5</th>
<th>F6</th>
<th>F7</th>
<th>F8</th>
<th>No. of Features</th>
<th>Feature used</th>
<th>Features (F 1 to 8)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Chat Forum Quiz Video Game</td>
<td>8</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Chat Forum Quiz Video Null</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Chat Forum Quiz Null Game Book Chat Forum Null Video Game</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Chat</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Chat</td>
<td>NuQuiz Video Game</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>App used</td>
<td>Assignment</td>
<td>Book Null Forum Quiz Video Game</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>App used</td>
<td>Assignment</td>
<td>Null Chat Forum Quiz Video Game</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>App used</td>
<td>Null</td>
<td>Book Chat Forum Quiz Video Game</td>
<td>7</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>App used</td>
<td>Null</td>
<td>Book Chat Forum Null Video Game</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>App used</td>
<td>Null</td>
<td>Book Chat Forum Quiz Null Game</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>App used</td>
<td>Null</td>
<td>Null Chat Forum Quiz Video Game</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>App used</td>
<td>Assignment</td>
<td>Null Chat</td>
<td>NuQuiz Video Game</td>
<td>6</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Figure 11: Table 4:
Rule No. Antecedent | Consequent | Support (%) | Confidence (%) | Lift |
--- | --- | --- | --- | --- |
1 | App_used | Assignment | 71 | 71 | 1.0 |
2 | App_used | Quiz | 72 | 72 | 1.0 |
3 | App_used | Forum | 92 | 92 | 1.0 |
4 | App_used | Game | 85 | 85 | 1.0 |

Figure 12: Table 5:

5 | App_used | Book, Forum | 73 | 73 | 1.0 |
6 | App_used | Game, Chat | 82 | 82 | 1.0 |
7 | App_used | Chat, Video | 82 | 82 | 1.0 |
8 | App_used | Forum, Chat, Video | 79 | 79 | 1.0 |
9 | App_used | Game, Chat, Video | 74 | 74 | 1.0 |
10 | App_used | Game, Chat, Forum | 76 | 76 | 1.0 |
11 | App_used | Game, Forum, Video | 73 | 73 | 1.0 |
12 | App_used | Video, Game, Chat, Forum | 71 | 71 | 1.0 |
13 | App_used | Video, Game, Book, Chat, Forum | 61 | 61 | 1.0 |
14 | App_used | Video, Game, Forum, Quiz, Chat, Book | 51 | 51 | 1.0 |
15 | App_used | Game, Assignment, Book, Forum, Chat, Quiz, Video | 43 | 43 | 1.0 |

Figure 13:

Pattern description | Min. support | Min. confidence | Number of patterns |
--- | --- | --- | --- |
7-itemset (7 feature items patterns) | 40% | 40% | 8 |
6-itemset (6 feature items patterns) | 40% | 40% | 28 |
6-itemset (6 feature items patterns) | 50% | 40% | 28 |
6-itemset (6 feature items patterns) | 60% | 40% | 8 |

Figure 14: Table 6:
7

Table 7:

<table>
<thead>
<tr>
<th>Exec. Records</th>
<th>20% Support</th>
<th>40% Support</th>
<th>60% Support</th>
<th>80% Support</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Apri FP BTPM</td>
<td>Apri FP BTPM</td>
<td>Apri FP BTPM</td>
<td>Apri FP BTPM</td>
</tr>
<tr>
<td>2500</td>
<td>387 195 161 141</td>
<td>121 110 118</td>
<td>113 101 117</td>
<td>103 92</td>
</tr>
<tr>
<td>5000</td>
<td>961 202 189 617</td>
<td>176 171 450</td>
<td>161 155 345</td>
<td>148 137</td>
</tr>
<tr>
<td>7500</td>
<td>1969 1255 349 1417</td>
<td>899 320 1117</td>
<td>750 305 917</td>
<td>559 291</td>
</tr>
<tr>
<td>10000</td>
<td>2946 1846 431 2014</td>
<td>1532 420 1403 1316</td>
<td>413 994 869 460</td>
<td></td>
</tr>
<tr>
<td>12500</td>
<td>3255 1850 454 2314</td>
<td>1714 445 1706 1505</td>
<td>431 1393 1110 41</td>
<td></td>
</tr>
<tr>
<td>15000</td>
<td>3779 2107 618 3513</td>
<td>1849 600 2889 1556</td>
<td>574 2419 1399 56</td>
<td></td>
</tr>
</tbody>
</table>

Figure 15: Table 7:

8

Component Description
Three nested iterations and an inner if statement with an assignment statement. But innermost iteration considers a low number of transactions well below N.

Procedure

1

Complexity
O(N*N*M+1+1), M is very close to 1 and very low to N

Procedure

2

Array create a statement, three nested iterations, inner if statement, and inner assignment statement. But innermost iteration considers a very low number of transactions well below N.

Complexity
O(1+1+N*N*M+1+1+1), M is very close to 1 and very low to N

Procedure

3

Two assignment statements within a for loop and one outside assignment statement.

Complexity
O(N*(1+1)+1) ~O(N)

Figure 16: Table 8:
A Novel Frequent Pattern Mining Algorithm for Evaluating Applicability of a Mobile Learning Framework

1. b) Results of the Proposed Novel Frequent Pattern Mining Algorithm

The proposed algorithm was implemented using the Python programming language. The following results were obtained after running the proposed algorithm on the dataset. Year 2023 © 2023 Global Journals

The algorithm’s efficiency can be improved further by the direct use of the binary incidence matrix format dataset. The other two algorithms used in this study for larger datasets. However, the proposed novel pattern mining algorithm shows better efficiency than the Apriori and FP-Growth algorithms. Meanwhile, in the applicability evaluation of the learning system, the proposed algorithm shows better efficiency than the Apriori and the FP-Growth for different support thresholds in various sizes of transactions. The proposed algorithm also shows the competitive value for the time complexity with the other two algorithms used in this study for larger datasets. However, the proposed novel pattern mining algorithm’s efficiency can be improved further by the direct use of the binary incidence matrix format dataset.

Meanwhile, our proposed algorithm gives 82% of the system applicability for a 75% threshold as the transaction’s minimum features. Finally, we can conclude that the proposed pattern mining algorithm provides accurate and more precise results for evaluating the proposed ML system’s applicability compared to the Apriori and FP-Growth algorithms. Meanwhile, in the applicability evaluation of the learning system, the proposed algorithm shows better efficiency than the Apriori and the FP-Growth for different support thresholds in various sizes of transactions. The proposed algorithm also shows the competitive value for the time complexity with the other two algorithms used in this study for larger datasets. However, the proposed novel pattern mining algorithm’s efficiency can be improved further by the direct use of the binary incidence matrix format dataset.


[Alzu’bi et al. ( )] ‘A Novel Recommender System Based on Apriori Algorithm for Requirements Engineering’.

S Alzu’bi , B Hawashin , M Elbes , M Al-Ayyoub . 2018 fifth international conference on social networks analysis, management and security (snams), 2018.


[Zhang and Zu ( )] ‘An Efficient Parallel High Utility Sequential Pattern Mining Algorithm’. C Zhang , Y Zu . 2019 IEEE 21st International Conference on High Performance Computing and Communications; IEEE 17th International Conference on Smart City; IEEE 5th International Conference on Data Science and Systems, (Zhangjiajie) 2019. HPCC/SmartCity/DSS.


[Wu et al. ( )] ‘Application of a two-step cluster analysis and the Apriori algorithm to classify the deformation states of two typical colluvial landslides in the Three Gorges, China’. X Wu , F B Zhan , K Zhang , Q Deng . Environmental Earth Sciences 2016. 75 (2) p . 146.


[Heaton ( )] Comparing dataset characteristics that favor the Apriori, Eclat or FP-Growth frequent itemset mining algorithms, J Heaton . 2016. 2016.
18 VII. CONCLUSION AND IMPLICATION


[Dunham (1)] Data mining: Introductory and advanced topics, M H Dunham . 2006. Pearson Education India.


[Pei and Han (1)] ‘Mining Frequent patterns without candidate generation’. J Pei , J Han . SIGMOD’ 00 Proceedings of the 2000 ACM SIGMOD international conference on Management of data, (New York, NY, USA) 2000.

[Han et al. (1)] ‘Mining frequent patterns without candidate generation’. J Han , J Pei , Y Yin . ACM sigmod record 2000. 29 (2) p. .

[Dolawattha et al. (1)] ‘Modelling the learner’s perspectives on mobile learning in higher education’. D Dolawattha , S Premadasa , P Jayaweera . 2018 18th International Conference on Advances in ICT for Emerging Regions, 2018.


[Soad et al. (1)] ‘Quality evaluation of mobile learning applications’. G W Soad , N F D Filho , E F Barbosa . 2016 IEEE Frontiers in Education Conference (FIE), (Èrie, PA, USA, USA) 2016.


[Doko et al. (1)] Sequential Pattern Mining Model to Identify the Most Important or Difficult Learning Topics via Mobile Technologies, E Doko , I A Bexheti , M Hamiti , B P Etemi . 2018. p. . (iJIM)
b) Results of the Proposed Novel Frequently Pattern Mining Algorithm


