Analyzing Student Enrollment Processes: A Process Mining Approach

***Lanka Pathmakumara **Gabbalage Dilshan ***Bathala Wicramasinhe ****Balasuriya Balasuriya *****Ponnahennadige Dias**

School of Engineering, Computing and Mathematics

University of Plymouth

Abstract

University student enrollment procedures are essential, but complexity and possible bottlenecks might reduce their effectiveness. In order to examine student enrollment data, this research investigates the use of process mining, more especially the Heuristics Miner algorithm. Event logs were used to identify the typical sequence of activities students go through during enrollment and to extract process knowledge. A visual process map that identified possible bottlenecks and offered insights into the enrollment process as a whole was produced as a consequence of the investigation.

The results imply that process mining provides colleges with useful instruments to increase enrollment effectiveness. Through the identification of bottlenecks and comprehension of process flow, establishments can optimize enrollment procedures, minimize processing durations, and ultimately improve the quality of student experience. The study admits its limitations with regard to the selection of performance metric and data coverage. It clears the path for further research that makes use of larger data sets, more data sources, and sophisticated process mining methods.

1.Introduction

In order for educational institutions to remain operationally effective while meeting the changing demands of their students, proper administration of student enrollments in LMS processes is critical. Conventional techniques for examining enrollment workflows frequently fall short in terms of the depth and accuracy required to spot anomalies, bottlenecks, and optimization opportunities. As a result, novel strategies that might offer detailed insights into the dynamics of enrollment processes are desperately needed.

In this study, a dataset including 141,713 entries and 31 columns from a student enrolling event log is analyzed using process mining techniques. The dataset records all of the activities that students do while they are enrolled, such as creating applications, updating their status, and maybe running into problems. Even though the data is useful, it needs to be handled carefully to ensure correct analysis because certain columns have missing values.

Finding the best routes for students to travel during the enrolling process and locating any bottlenecks that could lead to delays or irritation provide this situation's main challenges. These complicated sequences could be difficult for conventional data analysis techniques to adequately represent.

This research utilizes Heuristics Miner, a process mining algorithm, to analyze a student enrollment event log with the goal of improving efficiency and student experience. By uncovering the most frequent enrollment paths, identifying potential bottlenecks through dependency strengths, and evaluating process variations, we aim to streamline the enrollment process for both students and administrators.

The format of this document is as follows: Examining previous studies on process mining in educational contexts is the focus of the Literature Review section. The steps involved in pre-processing, using Heuristics Miner, and preparing the data are described in depth in the Methodology section. The consequences of the identified process model are examined in the Evaluation section. In conclusion, the section on Conclusions provides a summary of the major discoveries and suggests possible enhancements derived from the investigation.

2.Literature Review

Online assessment has become integral to modern education, serving various functions in both e-learning and blended learning settings [1]. Traditional data mining techniques have been widely applied to analyze educational data, including assessment data obtained from platforms such as intelligent tutoring systems and learning management systems [1]. However, these techniques often lack a comprehensive process perspective, focusing primarily on data dependencies rather than providing a holistic visual representation of the educational process [1].

In response to this limitation, process mining has emerged as a new research avenue, aiming to extract process-related knowledge from event logs recorded by information systems [2] [1]. Process mining offers a range of intelligent tools and techniques for discovering, analyzing, and improving processes, making it particularly suitable for educational data mining contexts [1] [2].

While the existing literature has predominantly focused on analyzing assessment data from online multiple-choice tests, there is a growing interest in extending process mining techniques to other educational data sources, such as student enrollment event logs [2] [1] [3] [4]By examining student enrollment processes through the lens of process mining, researchers can uncover valuable insights into enrollment patterns, student behavior, and the effectiveness of enrollment strategies [1] [2] [3] [4].

Research Paper 1 demonstrates the applicability of process mining, particularly utilizing the ProM framework, to educational data mining contexts. They analyze assessment data from online multiple-choice tests, employing techniques such as process discovery, conformance checking, and performance analysis [1]. By adapting these techniques to student enrollment event logs, researchers can gain insights into enrollment procedures, identify bottlenecks, and optimize the enrollment process for improved student experiences [1].

[2] complements this by focusing on the development of process mining techniques specifically tailored to educational data mining. They emphasize the importance of considering process-related knowledge in educational contexts and highlight the potential benefits of applying process mining to analyze student enrollment event logs [2].

Furthermore, [3] addresses issues related to authoring and personalization of online assessment procedures, providing valuable insights into the implications of different navigation strategies. Similarly, [4] emphasizes the importance of considering the process perspective in educational domains and proposes the development of tailored ProM plug-ins and authoring tools to facilitate the analysis of student enrollment event logs.

In conclusion, process mining presents promising opportunities for analyzing student enrollment event logs in educational contexts. By leveraging process mining techniques and tools like the ProM framework, researchers can gain valuable insights into enrollment processes, optimize enrollment procedures, and support personalized enrollment experiences for students [1] [2] [3] [4].

3.Methodology

3.1. Dataset description

The dataset tracks the status of student enrollment, whether the student is currently active, and various counts related to the case such as the number of times it has been reassigned, reopened, or modified. It also indicates whether the case follows the Service Level Agreement. The dataset represents a student's enrollment details with the following column details in Table 1.

Table 1-Column Description

Field Name	Description	Data Type
Case_ID	Unique identifier for each case or event.	String
Enrollment_Status	The status of the student's enrollment.	String
Active_Status	Indicates whether the student is currently active or not.	String
Reassignment_Count	The number of times the case has been reassigned.	Integer
Reopen_Count	The number of times the case has been reopened.	Integer
Modification_Count	The number of times the case has been modified.	Integer
SLA_Compliance	Indicates whether the case follows the SLA.	String
Student_ID	Unique identifier for each student.	String
Opened_By	The person who opened the case.	String
Opened_At	The date and time when the case was opened.	Date/Time
Application_Created_By	The person who created the application.	String
Application_Created_At	The date and time when the application was created.	Date/Time
Last_Updated_By	The person who last updated the case.	String
Last_Updated_At	The date and time when the case was last updated.	Date/Time
Contact_Type	The type of contact used for the case (e.g., email, phone).	String
Department	The department responsible for the case.	String
Enrollment_Category	The category of the enrollment.	String
Enrollment_Subcategory	The subcategory of the enrollment.	String
Issue_Description	A description of the issue related to the case.	String
Impact	The impact of the issue.	String
Urgency	The urgency of the issue.	String
Priority	The priority of the issue.	String
Assignment_Group	The group assigned to the case.	String
Assigned_To	The person assigned to the case.	String
Knowledge	Indicates whether the case is related to knowledge.	String
Priority_Confirmation	Indicates whether the priority has been confirmed.	String
Notification_Status	The status of the notification related to the case.	String
Closed_Code	The code related to the closure of the case.	String
Resolved_By	The person who resolved the case.	String
Resolved_At	The date and time when the case was resolved.	Date/Time
Closed_At	The date and time when the case was closed.	Date/Time

3.2. Data Preparation

3.2.1. Missing values

In this step, missing values are replaced with NA, Columns with more than 40% missing values were dropped, and missing values in specific columns were imputed or dropped based on context, and missing values for datetime columns are imputed with NA. Then, the time-related columns are converted to timestamps, and an event log is created.

3.2.2. Time Column Preprocessing

Time-related columns need to be converted to timestamps. In this step, the columns "Opened_At", "Application_Created_At", "Last_Updated_At", "Resolved_At", and "Closed_At" are converted to POSIXct objects. Then, they are combined into a single timestamp column.

3.3. Process Discovery with Heuristics Miner

The Heuristics Miner algorithm was employed as the primary process discovery technique. This algorithm analyzes the event log data to automatically discover the underlying process model. It identifies the most frequent sequences of activities and represents them as a process map. The process map visually depicts the enrollment journey, highlighting the typical flow of activities students' progress through.

3.4. Performance Analysis

Performance analysis was carried out in addition to process discovery to find any possible inefficiencies or bottlenecks in the enrolling procedure. Measures like the median processing time between tasks were computed to identify potential sources of delay. Understanding which enrollment process steps might need to be optimized was made possible by analyzing these performance indicators in conjunction with the process map.

4.Evaluation

Revealing the Enrollment Process to Students

We explore the findings from the process mining examination of the student enrolment event record in this section. The enrollment process's complexities were uncovered, and possible areas for improvement were noted, using the Heuristics Miner algorithm.

4.1. Data Preparation

To prepare it for process mining analysis, the data underwent multiple cleaning stages, as outlined in Section 3.2. (Data Preprocessing). Columns with a significant percentage of missing data (over 40%) were eliminated, and missing values were substituted with NA. For correct temporal analysis, datetime columns were transformed to timestamps in a standard format (UTC). Lastly, the remaining rows in the non-datetime columns that had missing values were eliminated.

4.2. Process Map Analysis

Based on the event log data, the process map created with process_map offers a visual depiction of the student enrolling procedure. Through the examination of node size and color (as determined by the median performance parameter), possible bottlenecks and places in need of development can be found.

4.2.1. Activities in the Enrollment Process

Every distinct activity found in the "Enrollment_Status" attribute of our data will be shown on the process map.

4.2.2. Identifying Bottlenecks:

Node Size: In a process map, larger nodes usually represent tasks that require a higher median time to finish as shown in the Figure 1. These can represent possible snags in the registration procedure.

Color of Nodes (taking into account median performance): If the default settings are applied, the color coding may indicate the median performance (time) of each task. In comparison to nodes with lighter colors, activities with darker colors may take longer on average to complete.

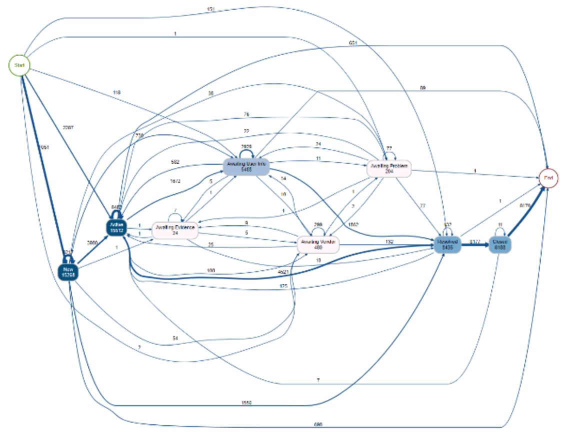

Figure 1-Process Map

4.3. Dependency Analysis

4.3.1 Dependency Matrix

A table displaying the likelihood (values between 0 and 1) of one activity leading to another is produced by the dependency_matrix function. For instance, the "New" row and "Active" column values of 0.999 show that a "New" enrollment nearly invariably becomes "Active."
It's in the Figure 2.

```
                   consequent
antecedent              Active Awaiting Evidence Awaiting Problem Awaiting User Info
  Active             0.9998821                 0        0.0000000          0.0000000
  Awaiting Evidence  0.0000000                 0        0.0000000          0.0000000
  Awaiting Problem   0.0000000                 0        0.9871795          0.0000000
  Awaiting User Info 0.0000000                 0        0.0000000          0.9996587
  Awaiting Vendor    0.0000000                 0        0.0000000          0.0000000
  Closed             0.0000000                 0        0.0000000          0.0000000
  End                0.0000000                 0        0.0000000          0.0000000
  New                0.9997488                 0        0.9743590          0.9986320
  Resolved           0.0000000                 0        0.0000000          0.0000000
  Start              0.9995629                 0        0.0000000          0.9915966
                   consequent
antecedent           Awaiting Vendor    Closed       End       New  Resolved Start
  Active                   0.0000000 0.0000000 0.9984663 0.0000000 0.9459867     0
  Awaiting Evidence        0.0000000 0.0000000 0.0000000 0.0000000 0.9090909     0
  Awaiting Problem         0.0000000 0.0000000 0.0000000 0.0000000 0.9871795     0
  Awaiting User Info       0.0000000 0.0000000 0.9888889 0.0000000 0.9994632     0
  Awaiting Vendor          0.9966667 0.0000000 0.0000000 0.0000000 0.9924812     0
  Closed                   0.0000000 0.9166667 0.9998776 0.0000000 0.0000000     0
  End                      0.0000000 0.0000000 0.0000000 0.0000000 0.0000000     0
  New                      0.9818182 0.0000000 0.9985694 0.9998783 0.9993553     0
  Resolved                 0.0000000 0.9998777 0.0000000 0.0000000 0.9924812     0
  Start                    0.0000000 0.0000000 0.0000000 0.9998582 0.9934211     0
attr(,"class")
[1] "dependency_matrix" "matrix"            "array"
```

Figure 2- dependency matrix

4.3.2. Causal Graph (Heuristics Net)

The enrollment procedure is represented graphically as a causal graph by the causal_net function. Activities are shown as nodes in this graph, with arrows indicating potential paths between them.
It's in the Figure 3.

4.3.3. Frequent Activity Sequences

We can determine the most common activity sequences by examining the causal graph or dependency matrix. A clear path indicating a typical enrollment path, for example, goes from "Start" to "Active" to "Awaiting User Info" to "Resolved".

4.3.4. Enrollment Activity Relationships

The relationships between enrollment actions are shown by the connected nodes in the causal graph and the high probability values (almost 1) in the dependency matrix. A clear transition from a new application to an active enrollment state, for instance, is suggested by a significant correlation between "New" and "Active".

4.3.5. Threshold

The code snippet also shows how to set the dependency matrix's threshold (0.95). This lessens clutter in the results and helps focus on the most likely activity transitions. It's in the Figure 4.

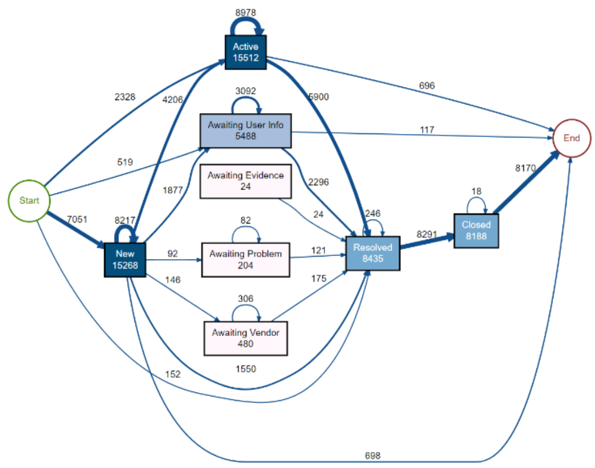

*Figure 3- **Heuristics Net***

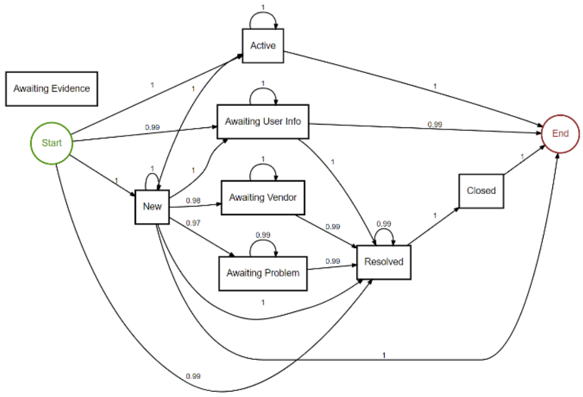

Figure 4- After Threshold ***Heuristics Net***

Dependency analysis helps discover opportunities for improvement, such as smoothing transitions and eliminating bottlenecks, by providing insights into enrollment flow. But there are restrictions. First off, broad applicability may be limited by data specificity. Second, using median performance indicators alone could miss subtleties in bottleneck diagnosis; percentiles or standard deviation could provide more insightful information. Finally, event log data may overlook elements like student behavior variances and human interventions, oversimplifying the intricacies of real-world registration. Even with these drawbacks, process mining is still useful for analyzing enrollment data. Recognizing these limitations emphasizes the need for additional studies using bigger datasets and cutting-edge techniques in order to properly understand the enrollment process.

4.4. Future Work

This study opens up new avenues for investigating student enrolling procedures. Future studies could look into how well Heuristics Miner works when used in conjunction with other process mining methods. Incorporating data sources other than the event log, like application details or student demographics, may also provide deeper insights into the variables affecting enrollment flow. Lastly, sophisticated methods such as compliance testing could be used to verify the identified process model and pinpoint variations that need more research.

5. Conclusion

This study looked into the use of the Heuristics Miner algorithm in process mining to examine student enrollment data. We were able to extract process knowledge and obtain important insights into the student enrolling path by utilizing event logs. Potential bottlenecks were highlighted by our investigation, which also produced a graphic depiction of the enrolling process that showed the average flow of events for students.

The research's conclusions have the potential to enhance the efficacy and efficiency of the student enrollment process. Universities can expedite the registration process, shorten processing times, and improve the overall student experience by locating bottlenecks and comprehending process flow.
It's crucial to recognize this study's limitations, though. The results' generalizability may be restricted by the particular data and performance metric selected. Furthermore, it's possible that the event log data underrepresents the complexity of registration procedures in the real world.

In the future, this research provides opportunities for more investigation. In the future, Heuristics Miner could be used to compare the efficacy of various process mining methods. Incorporating data sources other than the event log, like application details or student demographics, may also provide deeper insights into the variables affecting enrollment flow. Lastly, the process model that has been found may be validated and deviations that need more research might be found by using sophisticated techniques like conformance testing.

Universities may make the incoming student experience more efficient and friendly by regularly evaluating and enhancing their student enrolling procedures.

References

[1] W. Aalst and A. Weijters, “Process mining: A research agenda,” *Comput. Ind.*, vol. 53, pp. 231–244, Jun. 2004, doi: 10.1016/j.compind.2003.10.001.

[2] W. Aalst, C. Guenther, J. Recker, and M. Reichert, “Using Process Mining to Analyze and Improve Process Flexibility - Position Paper,” Jan. 2006.

[3] S. Srivastava and R. Bhatnagar, “A Study About Process Mining.” Rochester, NY, Feb. 23, 2019. doi: 10.2139/ssrn.3358228.

[4] M. Pechenizkiy and N. Tr, “Process Mining Online Assessment Data,” *Educ. Data Min.*, 2009.