PREDICTING STUDENTS PERFORMANCE BY ANALYZING THEIR BEHAVIOUR IN E-LEARNING SYSTEMS THROUGH INTERACTION PATTERN MINING

HIRA SALEEM

01-241202-018

A thesis submitted in fulfillment of the
requirements for the award of the degree of Master
of Science (Software Engineering)

Department of Software Engineering

BAHRIA UNIVERSITY ISLAMABAD

AUGUST 2023

APPROVAL FOR EXAMINATION

Scholar's Name: _HIRA SALEEM__ Registration No. __01-241202-018__

Program of Study: __MS (Software Engineering)__

Thesis Title: _Predicting Student Performance by Analyzing Their Behaviour in E-Learning Systems Through Interaction Pattern Mining

It is to certify that the above scholar's thesis has been completed to my satisfaction and, to my belief, its standard is appropriate for submission for examination. I have also conducted plagiarism test of this thesis using HEC prescribed software and found similarity index at 13% that is within the permissible limit set by the HEC for the MS degree thesis. I have also found the thesis in a format recognized by the BU for the MS thesis.

Principal Supervisor's

Signature:_____

Date: _____

Name:_____

AUTHOR'S DECLARATION

I, Hira Saleem hereby state that my MS thesis titled "Predicting Student Performance by Analyzing Their Behaviour in E-Learning Systems Through Interaction Pattern Mining" is my own work and has not been submitted previously by me for taking any degree from this university Bahria University Islamabad or anywhere else in the country/world.

At any time if my statement is found to be incorrect even after my graduation, the University has the right to withdraw/cancel my MS degree.

Name of scholar: Hira Saleem (01-241202-018)

Date: _____

PLAGIARISM UNDERTAKING

I, Hira Saleem , solemnly declare that research work presented in the thesis titled "Predicting Student Performance by Analyzing Their Behaviour in E-Learning Systems Through Interaction Pattern Mining" is solely my research work with no significant contribution from any other person. Small contribution/help wherever taken has been duly acknowledged and that complete thesis has been written by me.

I understand the zero-tolerance policy of the HEC and Bahria University towards plagiarism. Therefore I as an Author of the above titled thesis declare that no portion of my thesis has been plagiarized and any material used as reference is properly referred/cited.

I undertake that if I am found guilty of any formal plagiarism in the above titled thesis even after award of MS degree, the university reserves the right to withdraw/revoke my MS degree and that HEC and the University has the right to publish my name on the HEC/University website on which names of scholars are placed who submitted plagiarized thesis.

Scholar / Author's Sign: _____

Name of the Scholar: Hira Saleem (01-241202-018)

DEDICATION

To my beloved mother and father

ACKNOWLEDGEMENT

I would start by thanking ALLAH Almighty, with gratitude for giving me strength in every aspect of life and helping me in this thesis as well.

I wish to express my sincere appreciation to my thesis supervisor, Dr.Tamim Ahmed, for his support, guidance, and valuable feedback throughout this thesis. His expertise and encouragement have been instrumental in shaping my research and helping me to overcome the challenges that I faced.

I want to acknowledge my parents, who supported me and this thesis work is dedicated to my parents, who have been a constant source of support during the challenges of life and prayed for my success. I am truly thankful for the support in every aspect whether that is financial, emotional, or mental. I would like to thank my brother, Zeeshan Saleem and my friend Takreem-e-Fatima for their constant support that kept me motivated.

Lastly, I would like to acknowledge the contributions of all the participants who took part in my study, without whom this research would not have been possible.

ABSTRACT

This study focuses on interaction pattern mining within students' learning trajectories and addresses the impact of student interaction patterns on their performance in e-learning courses. Typically, instructors structure the course sequence based on their didactic and pedagogical strategies, with the intention of guiding students through their learning journey. However, in the absence of strict constraints, students might opt for learning paths that diverge from the predefined sequence. This context prompts an important question: What are the consequences for student learning outcomes when they pursue learning paths that deviate from the instructor's expectations? Within E-learning platform, students' interactions with course materials are logged as events. Employing Educational Process Mining techniques allows for the extracting statistical information, tracing and modeling of student actions during and Sequential Pattern Mining (SPM) used for sequential patterns within learning process.. We develop an LMS with which students can interact in both a directed and free manner. We utilized an event log containing 37,405 events, gathered from 76 undergraduate students. Prior to analysis, this log underwent a preprocessing phase. For experiment, we segmented log data into three distinct datasets. To derive statistical insights, we employed the PROM framework. Our investigation entailed the application of four distinct process discovery algorithms namely, Alpha Miner, Heuristic Miner, ILP Miner, and Inductive Miner also GSP algorithm were implemented through scripts based on the PM4PY library. The outcomes of our study revealed that students exhibited unique behaviors while accessing the LMS and engaging in activities. Interestingly, we observed that students who followed a predetermined sequence or interacted with the LMS in a guided manner achieved higher grades compared to their counterparts who navigated the LMS in a more random fashion.

TABLE OF CONTENTS

CHAPTER **PAGE**

APPROVAL FOR EXAMINATION ..... **ii**

AUTHOR'S DECLARATION ..... **iii**

PLAGIARISM UNDERTAKING ..... **iv**

DEDICATION ..... **v**

ACKNOWLEDGEMENT ..... **vi**

LIST OF TABLES ..... **xi**

LIST OF FIGURES ..... **xii**

LIST OF ABBREVIATIONS ..... **xiv**

CHAPTER 1 ..... **1**

INTRODUCTION **1**

1.1. Motivation 2

1.2. Research Gap 3

1.3. Problem Statement 5

1.4. Research Questions 5

1.5. Research Objectives 5

1.6. Outline of this thesis 6

CHAPTER 2 ..... **7**

LITERATURE REVIEW **7**

2.1 Interaction Patterns in E-learning 8

2.1.1 Moore's Theory of Interaction 8

2.1.2 Andersons' Interaction Equivalency Theory 10

2.2 Interaction Patterns in LMS 13

2.3 Process Mining for Online Interaction 14

2.3.1 Process Discovery 17
2.3.2 Process Conformance 17
2.3.3 Process Enhancement 20
2.4 Process Mining in Education 21
2.4.1 Alpha Miner 21
2.4.2 Heuristic Miner 22
2.4.3 Inductive Miner 23
2.5 Process Mining Tools 23
2.6 Sequential Pattern Mining for Online Interactions 25
2.7 Related Studies 25
CHAPTER 3 31
RESEARCH METHODOLOGY 31
3.1. Introduction 31
3.2. Proposed Methodology 31
3.3 System Design: 33
3.4 Data Collection 36
3.4.1 Log Data 37
3.5 Data Pre-processing 39
3.5.1 Data Cleaning 39
3.5.2 Noise Removal 41
3.5.3 Symbolization of operations 42
3.5.4 Data Transformation 42
3.5.5 Data Integration 43
3.5.6 Segmentation of Log Data 43
3.6 PROM Tool 44
3.7 PM4PY: 47
3.8 Application of Process Mining Algorithms 48
3.8.1 Alpha miner: 48
3.8.2 Heuristic Miner 50

3.8.3 ILP Miner Algorithm 52
3.9 Process Visualization 53
3.9.1 Petrinets 53
3.10 Conformance Checking: 53
3.10.1 Generalization 54
3.10.2 Precision 54
3.10.3 Fitness: 55
3.10.4 F1 Score 55
3.10.5 Simplicity 56
3.11 Application of Sequential Pattern Mining 56
3.11.1 Generalized Sequential Pattern 56
CHAPTER 4 59
RESULTS AND EVALUATION 59
4.1. Alpha Miner 59
4.2 Heuristic Miner: 61
4.3 Inductive Miner: 64
4.4 ILP Miner 67
4.5 Conformance Checking 69
4.6 GSP Results: 77
4.6.1 Student Count in each Cluster 78
4.6.2 Clusters of Students on basis of Performance 79
4.6.3 Visualization of Student Performance who follows Pre-defined Sequences 80
4.6.4 Visualization of student performance who follows random sequences 81
4.6.5 Student Performance comparison from both clusters 81
CHAPTER 5 83
CONCLUSION 83

LIST OF TABLES

TABLE NO. **TITLE** **PAGE**

Table 3-1 Attributes of student logs csv 39
Table 3-2 Student activities on LMS before data cleaning 40
Table 3-3 Student actions on LMS after data cleanining 41
Table 3-4 Symbolization of operations 42

LIST OF FIGURES

FIGURE NO.	TITLE	PAGE
Figure 2-1	Types of Process Mining	16
Figure 2-2	Four dimensional quality metrics	18
Figure 2-3	Process mining outcomes	20
Figure 3-1	Proposed methodology	33
Figure 3-2	System Interface	34
Figure 3-3	LMS features	34
Figure 3-4	Pre-defined sequence in LMS	35
Figure 3-5	Final grades csv	37
Figure 3-6	Student logs csv 1	38
Figure 3-7	Student logs csv 2	38
Figure 3-8	PROM Framework user interface	44
Figure 3-9	Import csv log file	45
Figure 3-10	CSV to XES conversion process	46
Figure 3-11	Student logs statistics in PROM	47
Figure 3-12	Alpha miner algorithm	49
Figure 3-13	Basic formulation of ILP miner algorithm	53
Figure 3-14	Generalization formula	54
Figure 3-15	GSP algorithm pseudo code	57
Figure 4-1	High performers Process model obtained using Alpha miner	59
Figure 4-2	High performers Petrinet obtained using Alpha miner	60
Figure 4-3	Low performers Process model obtained using Alpha miner	60
Figure 4-4	low performers Petrinet obtained using Alpha miner	61
Figure 4-5	High performer’s Process model obtained using Heuristic Miner	62

Figure 4-6 High performers Petrinet diagram obtained using Heuriatic Miner 62
Figure 4-7 Low performer's Process model obtained using Heuristic Miner 63
Figure 4-8 High performer's Petrinet diagram obtained using Heuristic Miner 64
Figure 4-9 High performer's Process model obtained using Inductive Miner 65
Figure 4-10 High performer's Petrinet obtained using Inductive Miner 65
Figure 4-11 Low performer's Process model obtained using Inductive Miner 66
Figure 4-12 Low performer's Petrinet obtained using Inductive Miner 67
Figure 4-13 High performer's Petrinet obtained using ILP Miner 68
Figure 4-14 Low performer's Petrinet obtained using ILP Miner 69
Figure 4-15 Conformance checking of high performers using simplicity 70
Figure 4-16 Conformance checking of high performers using Generalization 71
Figure 4-17 Conformance checking of high performers using F1 score 71
Figure 4-18 Conformance checking of high performers using Precision 72
Figure 4-19 Conformance checking of high performers using Accuracy 72
Figure 4-20 Summary of conformance checking of high performers 73
Figure 4-21 Conformance checking of low performers using simplicity 74
Figure 4-22 Conformance checking of low performers using Generalization 74
Figure 4-23 Conformance checking of low performers using F1 score 75
Figure 4-24 Conformance checking of low performers using Precision 75
Figure 4-25 Conformance checking of low performers using Accuracy 76
Figure 4-26 Summary of Conformance checking of low performers 76
Figure 4-27 Output of GSP algorithm 78
Figure 4-28 student count 79
Figure 4-29 Studemt performance based on interaction patterns 80
Figure 4-30 Normal distribution of high performer cluster 80
Figure 4-31 Normal distribution of high performer cluster 81
Figure 4-32 Comparison of performance of both clusters 81

LIST OF ABBREVIATIONS

LMS	-	Learning Management System
PM	-	Process Mining
EPM	-	Educational Process Mining
SPM	-	Sequential Pattern Mining
AM	-	Alpha Miner
HM	-	Heuristic Miner
ILP	-	Integrated Linear Programming
IVM	-	Inductive Visual Miner
ETM	-	Evolutionary Tree Miner
DFG	-	Directly Follow Graph
GSP	-	Generalized Sequential Pattern

CHAPTER 1

INTRODUCTION

In recent years, online learning has become prominently integrated into educational institutions worldwide. Within these virtual learning environments, educators design learning paths to facilitate students in acquiring the requisite competencies and skills relevant to their courses. Nevertheless, due to the diverse profiles of students and their distinct learning paces, as well as their varying interactions with heterogeneous media content, the ultimate trajectory of learning for each individual may deviate from the initially prescribed path within the learning management system (LMS).The learning footprints left by students within their digital learning environments possess potential for valuable insights, which can be utilize by teachers to enhance the learning experience. Regrettably, with the proliferation of large-scale education, the manual analysis of varied learning trajectories poses a formidable challenge for educators.

Learning Management Systems (LMS) have revolutionized the way courses are delivered and have become an integral part of online education. These platforms provide a wealth of data on student interactions, offering valuable insights into their learning behavior. By leveraging techniques such as process mining and sequential pattern mining, educators can uncover patterns in student interactions, understand their learning paths, and identify factors that influence their performance.

Process mining approach employed to visualize and analyze the learning process, along with the specific learning paths undertaken by students, with the ultimate goal of enhancing overall student outcomes. The proposed approach involves developing a comprehensive step-by-step modeling process that facilitates the application of sequential pattern mining techniques based on process mining principles. This methodology encompasses various stages, commencing with data collection, involving the identification of data sources, followed by data preprocessing, which involves

essential transformations to enable the implementation of the process mining techniques, is carried out. Additionally, data analysis is carried out, involving the use of process mining techniques. Last but not least, the results are displayed utilizing data visualization tools, emphasizing the process discovery analysis. By employing this Process Mining Model, educators and stakeholders can gain valuable insights into the learning journey of students, thus finding the way for comprehensive improvements in their academic performance.

1.1.Motivation

The data, which is collected at the system level within a university, includes information about students' high school records, course grades, and demographics [1]. These kinds of information are used to respond to questions about system-level issues including the rate of retention, percentages of graduates, and the length of time required to finish a degree.

Information collected at the individual level that is gathered through traditional educational assessments, such as performance on tasks related to learning, scores on specific test items, or overall achievement test scores. Data at this level have traditionally been thought of as the finest grain sizes used in teaching. Though more precise data has recently been added, thanks to technological developments in applications like Learning Management Systems (LMSs) [1].

This type of information, which includes all of the interactions that students have with a learning management system (LMS), has been described by researchers as "transaction-level data." These interactions include actions like submitting assignments, accessing lectures, and more. Such interactions have become valuable sources of information regarding learners' temporal preferences in engaging with learning tasks. They are often kept in log format, which makes it easier to do analysis pertaining to education and learning [1]. There are two dimensions: outcome measures and process

measures, which are relevant to learner interaction. Outcome measures assess the extent to which learners successfully accomplish a given task, whereas process measures focus on the actions and activities undertaken by learners during the task [1]. In the context of education, [2] identified two distinct categories of data metrics. The first group encompasses metrics that document the actual outcomes of learning, while the second group focuses on metrics that observe real-time interaction behaviours during the learning process, with uncertain correlations to actual learning outcomes. These process metrics capture interaction patterns, including detailed learner actions within Learning Management Systems (LMSs), considering factors such as frequency, time, and duration. The recorded interaction behaviours through log data encompassed various activities. These activities comprised reading texts, engaging with video lectures, interacting with multimedia content, seeking information, submitting assignments, and participating in discussion boards. In essence, the traceability of online interactions is a significant difference between interaction activities in LMS and traditional instructional environments. These interactions will be carefully monitored over time in LMSs, covering all aspects of the educational process. Learning management systems also offer a variety of restrictions based on different criteria, including participants, dates, hours, and activity categories. This granularity of data enables the identification of patterns closely linked to the structure and subject matter of online learning activities. Consequently, there exists substantial potential to enhance our understanding of interaction behaviour and, in turn, elevate the quality of online learning experiences. This paved the way for conducting our research within this research thesis.

1.2. Research Gap

The foundational concepts of process mining were initially introduced in 2011 [4]. It was subsequently proposed that process mining serves as the intermediary linking data science and process science [6]. Data science encompasses various fields, for

example machine learning, artificial intelligence and data mining, with the primary goal of extracting meaningful value from data. This value can take the form of data visualization, models and predictions, all of which provide insights to support decision-making processes. The activities involved in data science encompass extraction of data, preparation, exploration, transformation of data, and storage. On the other hand, process science is a discipline that combines knowledge from management and information technology. Instead of solely dependent on the facts and figures verified by the data, it uses an approach based on models to improve operational procedures. Rather, in-depth analysis or assessments of process phases are needed. [7]. Within the field of process mining, subjects such as workflow management, business process management, operations management, and process automation are explored [6]. The primary objective of process mining involves the extraction of insights from event logs within information systems. By employing diverse data analysis techniques and algorithms, the acquired information is extracted with the purpose of revealing, comprehending, monitoring, and improving processes[5].

Although there existed an extensive body of prior research concerning the application of Educational Process Mining (EPM), the literature primarily focuses on a limited set of algorithms, namely Evolutionary Tree Miner , Heuristic Miner, and Alpha Miner which provide feedback to the standard quality metrics in order to identify these problems. According to existing scholarly literature, the Inductive Miner (IM) algorithm for process discovery in educational datasets has yet to be explored [31] [33].

It is evident from the preceding discussion that data science especially process mining and interaction pattern mining can be applied to understand the behavioural pattern of students taught with the help of learning management systems. It is also pertinent to note that the students interacting with learning management systems (LMS) have no predefined patten to access the information. We do an interesting evaluation to see if students follow a predefined and a prescribed pattern would be more helpful or more productive for the students.

1.3.Problem Statement

The learning management systems are an integral part of education system especially in the aftermath of Covid-19. However, it is not known or investigated if a predefined or prescribed pattern advised to students would more assistive in their performance improvement or learning management. We evaluate, using process mining and interaction pattern mining techniques, if a prescribed interaction pattern for students be more productive.

1.4.Research Questions

RQ 1: How can we discover different learning trajectories taken by students while interacting with LMS?

RQ 2: How can we discover if a student followed a predefined sequence while interacting with LMS?

RQ 3: How can we compare the performance of students using a free interaction and a prescribed interaction pattern?

1.5. Research Objectives

Objective 1: Use Process Mining techniques on student logs extracted from LMS in order to discover hidden patterns.

Objective 2: Use Process Mining techniques on sequential data extracted from process mining models in order to discover hidden patterns.

Objective 3: Use k-means clustering to group students of similar behavior and compare performance of both clusters.

1.6. Outline of this thesis

The organization of this thesis is as follows: **Chapter 1** "Introduction" section includes the introduction of the study. **Chapter 2** "Literature Review" section which laid theoretical base for this research along with summarizes the previous works on Process mining and sequential pattern mining concepts and techniques in educational domain. **Chapter 3** "Research Methodology" section exhibits the conceptual framework used for this study and explains the working and methodology of the Process mining and sequential pattern mining for the extraction of pattern mining along with conformance checking of models and its implementation. **Chapter 4** "Results and Evaluation" section shows the inferences and results obtained. Finally, **Chapter 5** "Conclusion" section highlights our contributions, and future plans to extend this work.

CHAPTER 2

LITERATURE REVIEW

Learning Management Systems (LMS) have gained popularity in educational institutions as a means to support and enhance the learning experience. The vast amount of data generated through student interactions with LMS offers opportunities to analyze and understand student performance. Our study draws upon a comprehensive collection of literature that examines the utilization of process mining and sequential pattern mining techniques to uncover hidden patterns and gain insights into student performance within LMS environments, and categorized into four distinct groups, to establish its theoretical foundation. These groups encompass the following: 1) Interaction in e-learning, 2) Process mining in education, 3) Uncovering interaction or sequential patterns, and 4) Evaluating student performance. Moore's Theory of Interaction (1989) and Anderson's Interaction Equivalency Theory (2003) serve as the theoretical foundation for this investigation.

A course that is entirely delivered online, with no in-person interactions, shall be referred to as an online course for the purposes of this study. Self-report questionnaires were the main method used in many earlier research looking at student academic progress and retention. However, researchers are now able to avoid relying entirely on student reports of engagement because to the development and broad acceptance of learning management systems (LMS) equipped with tracking features. An additional source of data is made accessible to investigate the relationship between these behaviors and the successful completion of online courses by using interaction patterns as markers of online student behavior. Nonetheless, it is essential to understand the significance of fostering opportunities for student interaction within the e-learning environment. Such interaction

should not only encompass student-to-student and student-to-instructor communication but also encompass engagement with relevant and captivating course content.

2.1 Interaction Patterns in E-learning

In a study author claims that the availability of e-learning presents a solution that overcomes temporal and geographical constraints, enabling students to engage in studies at their preferred times. The rationale behind students opting for the online learning environment is attributed to the advantages of accessibility, flexibility, and convenience [8].

According to [9], interactions can be defined as reciprocal occurrences involving two objects and two corresponding actions. These interactions take place when the objects exert mutual influence on each other, particularly in the context of teachers and students. Nevertheless, the definition of interaction has been broadened to include how students interact with the course material in a classroom [10].

2.1.1 Moore’s Theory of Interaction

The fundamental theory essential for exploring interaction in e-learning is Moore's transactional distance theory [11]. In this theory, Moore proposed that distance is not solely a result of physical separation but a pedagogical phenomenon. Within this specific context, the author discerned three distinct categories of learner engagements. These categories encompass interactions between the learner and the learning materials, interactions between the learner and the instructor, and interactions amongst fellow learners [12]. Moore's three categories of interaction continue to be the most common and durable interaction classifications acknowledged by instructors, researchers, and

participants involved in online learning, despite the fact that researchers frequently discuss numerous other types of interaction from other perspectives [13].

2.1.1.1 Learner-Content Interaction

As articulated by (Moore, 1989), refers to the intellectual engagement between the learner and the educational material, leading to modifications in the learner's comprehension, perspective, or cognitive framework [10].

2.1.1.2 Learner-Instructor Interaction

Pertains to the bidirectional communication that occurs between the instructor and students within a course [11]. The central aim of the educator is to captivate and maintain the students' curiosity in the given subject, while also fostering motivation for learning and nurturing their self-direction and self-motivation [10].

2.1.1.3 Learner-Learner Interaction

Refers to the process of reciprocal communication between students, whether or not an instructor is present [10].

2.1.1.4 Learner-Interface Interaction

The concept of learner-interface interaction, as discussed by [10], can be used in both in-person and online learning settings. However, the majority of studies that made

use of this approach mostly looked at online education. Scholars have re-examined the original interaction theory and added new aspects of interaction to address the specific issues that have arisen in the setting of online learning [13].

One such dimension proposed by [14] is "learner-interface interaction." This type of interaction occurs between a learner and the technological tools employed to facilitate the online learning process. These mediation technologies include certain platforms, programmes, and lesson plans that make it easier for students to interact with the course material, teachers, and other students [14][15][16].

2.1.2 Andersons' Interaction Equivalency Theory

Building upon Moore's previous research concerning interaction, Terry Anderson developed the Interaction Equivalency Theory in 2003 (Anderson, 2003) [17]. This theory proposed that achieving deep and substantial formal learning is attainable as long as there is a strong emphasis on any one of three interaction modes: learner-instructor, learner-learner, or learner-content. As per Anderson's analysis, it is possible to provide the remaining two forms of interaction at reduced intensities or potentially exclude them, without undermining the holistic educational journey. However, other part of the theory proposes that a more fulfilling educational encounter is probable when higher degrees of any combination of these three interaction paradigms are accessible. Nonetheless, it should be acknowledged that such enriched experiences might not align with the efficiency in terms of cost or time, as observed in less interactive sequences. [17].

Anderson's observations indicate that prioritizing the interaction between students and the course content yields better performance when contrasted with the alternative forms of interaction. This observation gains supported in meta-analysis performed by [18], unveiling a direct correlation between well-structured course design components that facilitate diverse interaction modes and the resultant course achievements. Importantly, a marked enhancement in the magnitude of course outcomes is discernible,

specifically in instances where the emphasis is placed on fortifying the student-content interaction methodology.

In the 2013 International Conference on E-Learning, Terry Anderson demonstrates the advantages of learner-content interaction. He emphasized two specific attributes that make this form of engagement particularly attractive within the realm of higher education. The first pertains to its scalability, wherein educational material can be pre-recorded and subsequently accessed by a multitude of learners. This shift from personalized, one-on-one interactions to a model capable of accommodating a sizable student population underscores its cost-efficiency. Furthermore, the appeal of flexible learning, accessible at any time and place, is particularly attractive to individuals managing both familial and professional responsibilities. However, it is crucial to recognize that establishing meaningful engagement between learners and educational materials requires a notable degree of self-reliance and independent initiative, qualities that might be deficient in a considerable portion of the student body [19][20]. As a result, there emerges a necessity to cultivate behaviours that encourage self-directed and self-regulated education, ultimately augmenting student retention rates and overall academic achievements.

In order to explore the importance of the arrangement of a course in influencing patterns of interaction, a research study by [18]undertook a comprehensive analysis of various studies cantered around interaction methodologies. The term "interaction methodologies" pertains to the contextual settings or educational atmospheres established by instructors to stimulate distinct interaction patterns, as opposed to the direct interactions that are directly observed and documented. The findings of their investigation illuminated the central role played by these interaction methodologies in fostering meaningful engagements. In situations where the essential prerequisites for fostering a certain category of interaction are lacking, the manifestation of that particular interaction type becomes unfeasible. During the initial phases of online courses, the analysis of interactions between students and course content presented difficulties because of technological constraints. Nevertheless, the progress of Learning

Management Systems (LMSs) has alleviated this issue, enabling the monitoring of student-content interactions. This development has consequently introduced fresh prospects for scholarly investigations within the realm of online education [19].

In 2003, Thurmond did a thorough examination of earlier studies and produced a precise description for the idea of learner interaction in the context of e-learning [21]. According to his research, learner interaction describes how actively students engage with different elements of the online course, such as the course material itself, other students, the teacher, and the technology resources used in the learning process. Genuine interactions with these components include a two-way exchange of information; the primary aim is improving knowledge acquisition in the learning environment. Achievement of specified learning goals as well as fostering a deeper grasp of the course material is the ultimate purpose of these interactions [21].

The majority of existing studies have not explored the behaviour of interaction of at such a very low scale. Consequently, these studies are unable to address significant inquiries, such as the presence of sequential patterns in learner interaction within the realm of e-learning. Moreover, they fail to explore potential disparities in interaction patterns between different achievement groups. However, it might be difficult to distinguish between strong interaction patterns from low-achieving groups and ineffective ones from high-achieving groups without a thorough understanding of how learners interact in real-time at the micro-level. Additionally, assisting learners in achieving the ultimate goals of online courses and improving general learner performance require focus. In order to understand the fundamental dynamics of how learners interact in this setting, our work is focused on assessment and analysis of behaviour of student in e-learning environment.

In order to focus on the actual interaction and investigate behaviour of e-learning engagement, it is essential to possess data that reveals temporal and sequential traces of interactions. Fortunately, Learning Management Systems (LMSs) offer a wealth of such data, encompassing real-time information on learner interactions in a sequential manner.

2.2 Interaction Patterns in LMS

Chung (2014) offers three specific kinds of analysis that are useful in understanding student performance in order to explore the procedure of teaching and learning. This data, which is collected at the system level within a university, includes information about students' high school records, course grades, and demographics [1]. These kinds of information are used to respond to questions about system-level issues including the rate of retention, percentages of graduates, and the length of time required to finish a degree.

The second category consists of information at the individual level that is gathered through traditional educational assessments, such as performance on tasks related to learning, scores on specific test items, or overall achievement test scores. Data at this level have traditionally been thought of as the finest grain sizes used in teaching. Though more precise data has recently been added, thanks to technological developments in applications like Learning Management Systems (LMSs) [1].

This type of information, which includes all of the interactions that students have with a learning management system (LMS), has been described by researchers as "transaction-level data." These interactions include actions like submitting assignments, accessing lectures, and more. Such interactions have become valuable sources of information regarding learners' temporal preferences in engaging with learning tasks. They are often kept in log format, which makes it easier to do analysis pertaining to education and learning [1].

Chung (2014) presented two crucial dimensions: outcome measures and process measures, which are relevant to learner interaction. Outcome measures assess the extent to which learners successfully accomplish a given task, whereas process measures focus on the actions and activities undertaken by learners during the task [1].

In the context of education, [2] identified two distinct categories of data metrics. The first group encompasses metrics that document the actual outcomes of learning, while the second group focuses on metrics that observe real-time interaction behaviours

during the learning process, with uncertain correlations to actual learning outcomes. These process metrics capture interaction patterns, including detailed learner actions within Learning Management Systems (LMSs), considering factors such as frequency, time, and duration.

In the study conducted by [3], the recorded interaction behaviours through log data encompassed various activities. These activities comprised reading texts, engaging with video lectures, interacting with multimedia content, seeking information, submitting assignments, and participating in discussion boards. In essence, the traceability of online interactions is a significant difference between interaction activities in LMS and traditional instructional environments. These interactions will be carefully monitored over time in LMSs, covering all aspects of the educational process. Learning management systems also offer a variety of restrictions based on different criteria, including participants, dates, hours, and activity categories. This granularity of data enables the identification of patterns closely linked to the structure and subject matter of online learning activities. Consequently, there exists substantial potential to enhance our understanding of interaction behaviour and, in turn, elevate the quality of online learning experiences.

2.3 Process Mining for Online Interaction

The foundational concepts of process mining were initially presented by Wil van der Aalst in his publication titled "Process Mining: Discovery, Conformance, and Enhancement of Business Processes," which was published in 2011 [4]. In the following year, van der Aalst, together with Adriansyah, de Medeiros, and other scholars, put forth the "Process Mining Manifesto" to advocate for the field of process mining and laid its basic foundation, rules, and obstacles. These definitions continue to hold significant influence in the scientific literature[5].

[6] Proposed that process mining serves as the intermediary linking data science and process science. Data science encompasses various fields, for example machine learning, artificial intelligence and data mining, with the primary goal of extracting meaningful value from data. This value can take the form of data visualization, models and predictions, all of which provide insights to support decision-making processes. The activities involved in data science encompass extraction of data, preparation, exploration, transformation of data, and storage. On the other hand, process science, as defined by the same author, is a discipline that combines knowledge from management and information technology. Instead of solely dependent on the facts and figures verified by the data, it uses an approach based on models to improve operational procedures. Rather, in-depth analysis or assessments of process phases are needed. [7]. Within the field of process mining, subjects such as workflow management, business process management, operations management, and process automation are explored [6].

The primary objective of process mining involves the extraction of insights from event logs within information systems. By employing diverse data analysis techniques and algorithms, the acquired information is extracted with the purpose of revealing, comprehending, monitoring, and improving processes[5].By utilizing the increasing computational capabilities available today, process mining enables comprehensive visualizations that capture both complex and straightforward process flows. These visualizations provide valuable insights, allowing users to explore processes from basic and aggregated perspectives to more intricate and detailed views, thereby enhancing our overall understanding of all processes [7].

The "Process Mining Manifesto" [5] introduced three key attributes aimed at enhancing the comprehension of process mining methodologies. It first highlights the fact that process mining goes beyond simple process discovery. Figure 2-1 clearly illustrates how process discovery coexists with conformity testing and process enhancement in addition to being one of the main approaches inside process mining. [6]. Secondly, process mining should not be considered a sub discipline of data mining. While data mining techniques can be utilized, most of them do not focus on processes, necessitating

the development of novel algorithms. Lastly, the third characteristic underscores that despite process mining's extraction of knowledge from historical data, it extends beyond offline analysis. The results derived from process mining can be effectively applied to current cases, allowing for real-time decision-making and utilization [5].

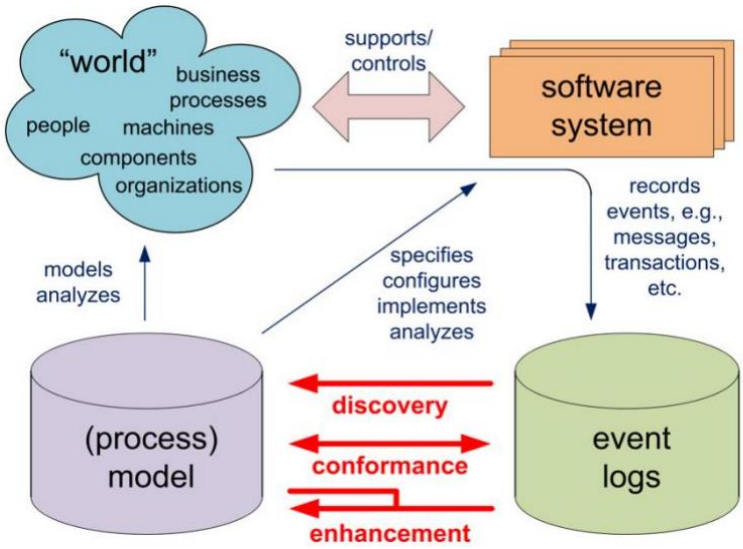


Figure 2-1 Types of Process Mining

As depicted in Figure 2-1, the practice of process mining necessitates the utilization of a software system for the storage of event data. This software program is essential for enabling and controlling actual procedures. Additionally, the student logs were collected, preprocessed and lastly analyzed by the process mining tool, The objective of such analysis is to executing various tasks including discovery, compliance, or enhancement [22]

Numerous case studies of process mining field were employed an established approaches and algorithms of process mining, which are conveniently accessible in ProM. Among these algorithms, the Fuzzy Miner holds the predominant usage share of 24%, followed by the Heuristics Miner at 16% [23].

2.3.1 Process Discovery

Process discovery constitutes the initial phase among the three fundamental categories within the realm of process mining. It looks for concealed trends and patterns within the collection of events with the intention of obtaining information from log data that was extracted from LMS. To create a process model that covers and explains what has been observed in the data is its main goal. This approach does not rely on any pre-existing model or prior information, as indicated in Figure 2-1. The intention is to portray the actual scenario rather than an idealized or subjectively documented one. Although other viewpoints, such as a social network, can also be taken into account, the final model is often described in the Petri net or BPMN notation. Furthermore, process discovery is commonly employed as an initial stage for subsequent practices and analyses.

2.3.2 Process Conformance

Conformance checking is the second method used in process mining. With this technique, log data from LMS connected with similar approach is compared with a process model that has already been created. By doing so, it aims to identify any disparities between the actual process and the modelled process [5]. Conformance checking serves the purpose of detecting, localizing, and explaining any deviations that occur, while also quantifying and measuring their severity [6]. Similar to the illustration provided in Figure 2-2 [5], this technique requires an event log and a pre-established model as inputs. As a result, it generates a diagnostic analysis that highlights the differences or similarities between the model and the recorded data [5].

It is very important to employ established quality standards when assessing the alignment of process models derived through the utilization of process mining techniques. Generalization, simplicity, fitness and precision are the four often used major quality metrics exists currently in literature , illustrated in Figure 2-2 as described in [24].

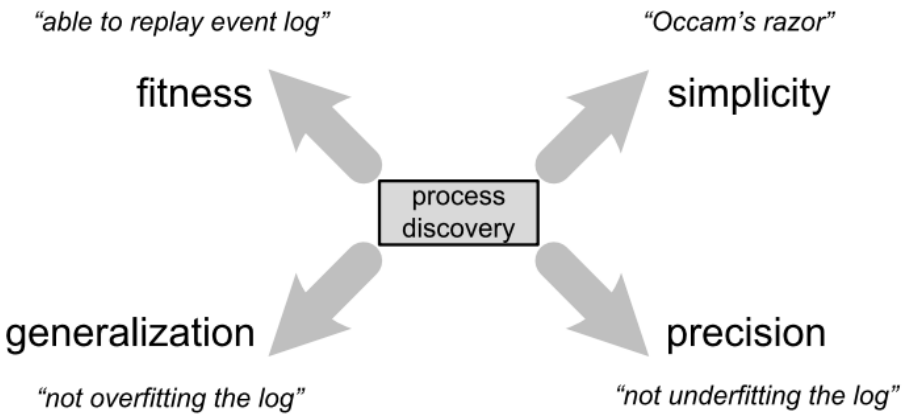


Figure 2-2 Four dimensional quality metrics

2.3.2.1 Fitness

Fitness denotes the proportion of patterns within an established procedural framework that can be reproduced from the recorded sequence of events." It measures, in more straightforward terms, how closely the process model can mimic the observed behaviour in the event log. Therefore, a fitness score of 1.0 denotes that the process model can replicate every trace in the event log, accounting for a significant amount of the observed behaviour. [25].

2.3.2.2 Simplicity

Simplicity consider model's interpretability, readability along with its complexity [26]. It displays the simplest structure which seems consistent with the behaviour which is being observed. [27].

2.3.2.3 Precision

Precision is the measure of the degree to which the process model aligns with the observed behaviour as documented in the event log. A model that exhibits high precision effectively guards against the occurrence of under fitting."[28]

2.3.2.4 Generalization

Generalization plays a crucial role in establishing the level of abstraction within a process model. It serves as an indicator of whether the process model comprehensively represents the entirety of process instances or merely encapsulates the discerned behavioural patterns, thereby signifying a notably specific nature.[25].

[29] Conducted a study examining few crucial aspects of the process conformance strategy. These attributes encompass time measurements, data regarding individual cases, details about available resources, as well as explicit and implied methodologies involving multiple occurrences with diverse sequences. They also encompass support for the lifecycle of activities, restrictions related to numerous instances, identification and correction of deviations from regulations, the analysis and handling of violations, analysis of the root causes of such violations, and the quantification of the degree of adherence or deviation from established norms. [30]

In the realm of conformance checking techniques, two methods have emerged as notable in academic literature: the Conformance Checker and the Linear Temporal Logic (LTL) Checker. Instead of contrasting a model with the log, the former determines if the event logs adhere to particular LTL formulae by contrasting a set of requirements against the LTL temporal logic. The Conformance Checker, nonetheless, necessitates both a model and an event log to enable replay within a Petri net model, all the while acquiring diagnostic information. [31].

2.3.3 Process Enhancement

Process extension or augmentation is the third category for process mining. This methodology's main goal is to improve or expand an existing process model using data from the event logs of a real-world process [32]. Consequently, the input for process enhancement consists of an event and an established model. The anticipated outcome is a more comprehensive or refined model, as depicted in Figure 2-3 [5].

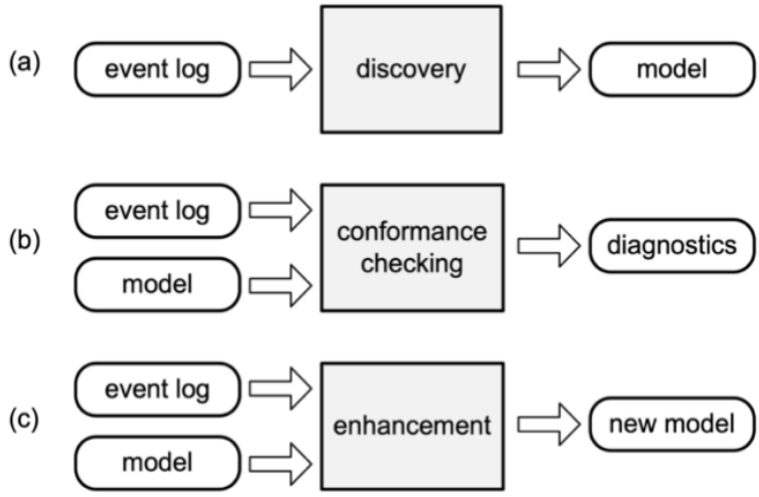


Figure 2-3 Process mining outcomes

Two different types of improvement are facilitated by process enhancement. The first type entails fixing the model to ensure a more accurate representation of reality. Integrating a novel perspective in a model and comparing it with the event log constitute elements of the second category of analysis [5].

[30] Outlined several potential processes and methods for process improvement. Some of these are the extension of models, using predictive methods, adopting an organizational viewpoint, dealing with concept drift, utilizing decision mining, and optimizing resource allocation.

2.4 Process Mining in Education

Education Process Mining (EPM) constitutes an integral aspect of the technique employed within Educational Data Mining (EDM). While EDM concentrates on data-centric analysis, EPM, on the other hand, centers on process-centric analysis. Its primary purpose involves scrutinizing and uncovering genuine learning processes and behaviors by harnessing knowledge extracted from event data. Drawing from the principles of process mining, EPM possesses the capacity to generate process models aligned with specific analysis objectives, thereby enhancing the comprehension of learning processes.

The implementation of Educational Performance Management (EPM) has been observed in educational settings. However, it exhibits a limitation when deal with complex learning processes. The generated model struggles to adequately represent the overall learning behavior and fails to offer explicit models in case of complex and extensive learning processes. To address this drawback of EPM, researchers have introduced clustering methods aimed at enhancing the resulting models [33].

2.4.1 Alpha Miner

An extensively utilized technique within the field of process mining is the alpha miner, also referred to as the α-algorithm [33]. This algorithm, identified as the initial discovery algorithm, laid the foundation for subsequent advancements in the field [6]. Its primary objective revolves around bridging the gap between gathered event log data and the discovery of process models. However, it is important to note that the original alpha miner algorithm possesses a prominent drawback as it does not incorporate frequencies, the management of data with inherent inconsistencies, the inability to unearth duplicated and concealed tasks, the handling of loops characterized by lengths of one or two, thereby failing to ensure reliability [33]. Consequently, its applicability is confined to event logs

that are free from noise, which is a relatively uncommon occurrence in practical learning datasets [34]. These constraints are tackled in subsequent iterations of the algorithm: the α+ (alpha plus) algorithm demonstrates proficiency in managing concise loops, the α++ (alpha plus plus) algorithm extends its capacity to accommodate intricate patterns within the procedure, and the α# algorithm excels in unveiling latent and unnoticed tasks [33].

After analyzing the event log α-algorithm transforms it into a Petri Net representation, connecting the activities in a way that represents their causal relationships. The resulting model encompasses all the places, transitions, and arcs determined during the algorithm's execution [35].

2.4.2 Heuristic Miner

Heuristic Miner is a process mining algorithm developed by Dr. Tonne Weitjers, uses a heuristic methodology, which is a method of teaching that involves presenting data and making inferences from it. The Alpha algorithm has flaws, such as invisible task, non-free-choice, implicit place, and length-one-loop which are addressed by the heuristic approach. Heuristic miner representation is same as casual nets. From existing process models, the control flow perspective is mined by this algorithm. In event log, noise can be handled by heuristic miner algorithm and it can also display the important behaviour of a process model even when not all details and exceptions are displayed.

Heuristic mining algorithm exhibits three notable advancements in comparison to the Alpha Algorithm. Firstly, it incorporates considerations of frequencies and significance, enabling the filtration of noisy or infrequent behaviour. Consequently, this feature renders it less susceptible to noise and incomplete logs [36]. Secondly, it possesses the capability to identify and detect short loops within the data. Thirdly, it permits the exclusion of individual activities from the analysis. Nonetheless, it is important to note that the algorithm does not ensure the generation of sound educational process models [34].

The Heuristic Miner algorithm employs a probabilistic approach, determining the frequencies of task relations such as causal dependencies and loops. It then proceeds to generate tables and graphs based on this dependency and frequency data. Notably, this algorithm exhibits a commendable resilience to both noise and incomplete information present in the logs [35].

2.4.3 Inductive Miner

Inductive visual miners (IvMs) remain an underexplored area in the realm of research, particularly concerning educational datasets [33]. Although there existed an extensive body of prior research concerning the application of Educational Process Mining (EPM), the literature primarily focuses on a limited set of algorithms, namely Evolutionary Tree Miner , Heuristic Miner, and Alpha Miner which provide feedback to the standard quality metrics in order to identify these problems. According to existing scholarly literature, the Inductive Miner (IM) algorithm for process discovery in educational datasets has yet to be explored [31]. Numerous process mining algorithms have been proposed, yet none of them consistently produce high-quality metrics in all scenarios. However, the Inductive Miner algorithm has gained significant popularity in the business domain due to its promising outcomes [37]. The Inductive Miner represents advancement over the Alpha and Heuristics miners, facilitating the exploration of event logs. It possesses the ability to handle infrequent patterns of behaviour and process large event logs while guaranteeing soundness [38].

2.5 Process Mining Tools

Although process mining has a relatively concise historical background, a diverse range of software solutions has already been developed to facilitate the implementation

of methodologies from process mining field. Notably, the ProM Framework, Disco, and RapidProM have surfaced as the three predominant and firmly established choices within this toolset [39].

In another study [23] discussed that numerous commercial process mining tools have emerged within the field, including ARIS Process Performance Manager, Celonis Process Mining, XMAnalyze, Disco, ProcessAnalyzer, Discovery Analyst, and Interstage Process Discovery. Among these tools, ProM stands out as a comprehensive process mining environment, being the tool of choice in approximately 84% of cases within the realm of Enterprise Performance Management (EPM).

The ProM Framework is an open-source tool designed to address the challenge posed by different tools employing their own formats for the interpretation and retention of data, which produces inconsistent results and makes result comparisons difficult [40]. ProM distinguishes itself as a versatile and adaptable tool which incorporates various methodologies of process mining which organized as plug-ins. It integrates numerous existing functionalities, supports multiple formats and languages, and enjoys widespread application and recognition [41].

Disco, the second tool, offers a distinct advantage in its emphasis on a visually appealing interface that promotes user-friendly functionality, allowing for filtering options in event logs [39].

The third tool, RapidProM, involves the development of process mining workflows by combining RapidMiner, platform utilized for formulating, executing and analysis of data and its resolutions, using ProM tool and integrated add-ons. [39]. Although Disco is regarded as more accessible and practical, ProM remains the most popular tool among those considered [42]

2.6 Sequential Pattern Mining for Online Interactions

Sequential pattern mining (SPM) constitutes a data mining approach crafted to discern patterns within consecutively ordered sets of items. Initially introduced to explore pattern discovery in customer purchase sequences, this technique has gained increasing attention in the context of e-learning and educational data mining. Consequently, SPM has found application in the field of education, serving to enhance and optimize teaching and learning processes through technology integration [43]. In a study [44], author revealed that sequential pattern mining (SPM) has the capability to unveil concealed patterns of ordered events that exhibit noteworthy characteristics, for example being frequent among achieving better grades students but infrequent among those who achieved low grades. Further [45] discuss five extensive classes of sequential pattern-mining algorithms, which include, Breadth First Search-based strategy, Depth First Search strategy, Apriori-based algorithm, sequential closed-pattern algorithm, and incremental pattern mining algorithms.

Within the domain of sequential pattern mining, two primary methodologies are commonly employed: such as pattern growth-based and apriori-based techniques. The foundational algorithms used for this purpose, such as GSP and AprioriAll are rely on the Apriori property, originally introduced in association rule mining. A set of data projection-based algorithms, including FreeSpan and PrefixSpan were later proposed [46].

2.7 Related Studies

[47] Describes the results of identifying and analysing the educational trajectories of students within an introductory programming course using PM and SPM approaches. The results demonstrated how each student can act differently, resulting in a variety of sequences.

In the study conducted by [36] the utilization of data clustering was employed with the aim of enhancing the precision of Process Mining models pertaining to student behavior.

In a separate study, [48] employed Process Mining methodologies to investigate and evaluate the learning patterns of students in Massive Open Online Courses (MOOCs), distinguishing between those that yielded successful outcomes and those that did not.

[26] Discusses the use of educational process mining to track students' learning paths in e-learning courses and assess the impact of their choices on their learning outcomes.

[49] Conducted a weekly assessment of student behavior for the duration of a semester. The Learning Management System (LMS) log data was collected for this study, and process mining techniques were applied. The primary objective of their research was to develop a more effective learning strategy for students. The results of their investigation revealed that the implementation of systematic teaching strategies yielded a significant influence on both student engagement and academic performance.

In study [50], Process Mining methodologies were employed to scrutinize Learning Management System (LMS) logs obtained from Moodle, with the aim of investigating the concurrence between teachers' pedagogical plans and students' actual utilization of LMS resources. The findings of this analysis indicated a misalignment between students' behaviors and the intended pedagogical objectives expected by the teachers.

The integration of clustering and sequential pattern mining techniques used in [51] to analyze the log data using the generalized sequential pattern (GSP) algorithm. The core purpose behind the utilization of this technique was to identify patterns and trends that could highlight the factors contributing to academic success or failure among two distinct groups of students.

In their recent study, [34] employed a novel Process Mining (PM) algorithm called the Inductive Miner (IM) to investigate and uncover learning processes within the Learning Management System (LMS) Moodle. The primary aim was to utilize the IM

technique for learning model discovery, thereby offering a potential means to avert instances of learning failure in the LMS environment. This innovative approach holds promise in enhancing educational outcomes and optimizing the learning experience for students engaging with Moodle.

[52] Discusses a proposed approach for the analysis of students' learning behavior in LMS, based on their profiles and uses principles from process mining and graph theory and aims to provide effective visualization and information about students' behavior.

The study [53] uses process mining approach inductive miner algorithm to reveal students' interaction profiles and knowledge acquisition in e-learning courses. It provides insights on the effectiveness of a learning design and types of student interactions in an online course environment.

In [54] researchers used process mining, specifically the Heuristic Miner algorithm, to compare student learning patterns between programming courses and non-programming courses. The results show that the process model can represent the event log well. The modeling process also shows the advanced behavioral appropriateness and degree of model flexibility of the two subjects.

The study [55] analyzed online course data from high and low grade groups using Fuzzy Mining algorithms to differentiate their behavioral patterns. The results suggested that students who spent more time had higher grades during the curriculum.

The researchers in [56] propose a method that provides a systematic approach to analyze programming learning history based on the learning process using the sequential pattern mining algorithm. It allows for the extraction of frequent patterns that help to further understand the learning process of programming students.

The study [57] developed an e-learning platform capable of detecting sequential actions in students' behaviors to transform their learning process into a productive one.

The study [58] evaluated student satisfaction and interaction with a Learning Management System (LMS) and found a positive attitude towards its use, with an overall satisfaction level of 80.07%. However, online interactivity needs improvement.

The study [59] aims to explore the deviation between learning design and actual execution in LMS and presents the use of process mining in LMS with regards to this deviation and provide three perspectives on approaching student behavior towards the deviation between learning design and actual execution.

Previous studies [60][61] discuss process mining as a research discipline and its relevance in bridging data mining and business process modeling. It explains the different types of process mining, including process discovery, conformance checking, and enhancement.

The study [43] aimed to investigate the navigational patterns of students on Moodle. The results suggest that sequence modeling can yield interesting patterns that provide insights into students' engagement and use of learning resources.

In study [62] researchers explored the online behavior of undergraduate students in a blended statistics course using Moodle. Data mining techniques including sequential pattern mining were applied to uncover common access patterns and resource preferences, providing insights for improved course design.

The study [63] analyzes student data in Moodle courses using clustering techniques along with PROM tool. The approach distinguishes different variables and is shown to be effective in identifying at-risk students and providing feedback to instructors.

The study [64] used a process mining methodology to analyze the educational trajectories of 794 engineering students, and provides useful insights for teachers to improve the curriculum design and support programs for minority students.

The study [65] analyzed students' behavior in an online course using educational process mining and found that students who watch course videos first and attempt quizzes before the hard deadline are more likely to pass the course.

The researchers in [66] analysed student behaviour using web tracking and process mining tools to pinpoint areas of the tutorials that would need more explanation or modification.

In study [67], The scholars proposed an approach rooted in sequential pattern mining to examine the process of implementing work-with-free-response (WFR). The

outcomes revealed recurrent patterns, facilitating the recognition of typical errors committed by students.

The study [68] Presents an interactive and encompassing structure for assessing the impact of student learning through the utilization of process mining techniques. This framework scrutinizes user actions and facilitates digital data compilation, resulting in perceptive deductions concerning both student learning outcomes and patterns of engagement.

In study [41] Scholars introduce a process mining methodology that incorporates clustering methodologies to enhance the accuracy of inferred process patterns. Two specialized ProM add-ons are developed to facilitate the extraction of precise process models, and this novel technique is demonstrated through the analysis of a practical real-world scenario.

The study [33] analyzed the students' behavior, their preferences on learning subjects, and their interactional behaviors during class using process mining techniques including Directly Follows Visual Miner (DFVM) and its variant named Inductive Visual Miner (IVM). Effective actions can be taken by teachers and administrators based on this analysis to motivate students to attend and improve their understanding of the topics.

The study [69] aimed to enhance learning efficiency for personalized learning by extracting insights from students with different learning styles by combining educational data mining with process mining techniques.

The study [70] explains the possibilities, difficulties, and viability of mining educational processes. The primary emphasis revolves around the exploration and examination of social networks, employing a clustering methodology to deconstruct educational procedures, alongside the utilization of key performance indicators.

This study [71] Presents a methodology that integrates process discovery, conformance assessment, and performance analysis with user-friendly process models structured around directly followed models. These innovative methods were deployed to various bussiness workflows within a department of the Queensland Government. This

implementation yielded dependable revelations and enabled precise identification of deviations from established procedural norms.

[46] Discuss the use of sequential pattern mining in Educational Data Mining to identify course trajectories leading to academic success and to discover specific courses that may influence students to drop out and present preliminary results demonstrating the usefulness of sequential pattern mining in solving problems related to education.

[72] Discuss the use of process mining techniques to analyze data generated from Massive Open Online Courses (MOOCs), through which instructor analyzing students' learning habits, video watching behavior, in order to provide insights into their interaction pattern and its relation to their performance.

The study [73] analyzes the behavioral clustering of students in a course and mined generated sequence of log data using sequential pattern mining to detect differences between passing and failing groups, and identify points of disengagement for potential failure indicators.

Researchers in [74] discuss the use of process mining on curriculum data to improve the design of a curriculum and provide recommendations to students based on expected outcome.

The study [75] utilized process mining to understand the learning strategies of students in an LMS-based computing course. The findings reveal that efficient learners engaged more in reading-based preparatory activities.

[76] Analyze student learning behavior patterns using sequential pattern mining to examine both high- and low-achieving students.

The study [77] investigated learner's behavior in a virtual learning environment using process mining. The results show that process mining can reveal valuable insights into learning behavior and can assist educators in designing effective learning strategies.

The study [78] examined the use of process mining to analyze patterns of student behavior in an online course. The results indicated that process mining can provide meaningful insights for improving the course and enhancing student learning outcomes.

CHAPTER 3

RESEARCH METHODOLOGY

3.1. Introduction

The analysis of student performance based on their interactions with Learning Management Systems (LMS) using process mining and sequential pattern mining techniques requires the implementation of robust methodologies and approaches. These methodologies and approaches encompass various stages, from data preprocessing to the process discovery and from sequential patterns to performance analysis. This section explores the key methodologies and approaches employed in the analysis of student performance within LMS environments, aiming to provide insights into how data is prepared, integrated, and analyzed to derive meaningful conclusions.

3.2. Proposed Methodology

Our research methodology involves the implementation from 3 major techniques including interaction pattern mining, process mining and sequential pattern mining which further including various steps which we will elaborate in detail in this section.

First we develop Learning management system with which student can interact in guided or free manner, when students interact in restricted or guided mode they follow a predefined path by instructor which is watch (video lecture, download handouts and attempt quiz) in a row, otherwise they are free to access it randomly. We analyze interaction patterns of both groups in order to figure out the impact of following path defined by instructor on their performance.

After collecting the student log data and their final grades into csv files from Learning Management System (LMS), preprocessing begins, and we eliminated missing values, duplicate and irrelevant actions, we perform symbolization of operations, transformation of data and integration, we finally did segmentation of data into 3 clusters based on their final grades using k-means clustering which includes high performers, low performers and all students clusters.

For statistical analysis we used PROM framework tool, which provide summary and desired information of student logs.

Then we applied process discovery algorithms on each dataset. We use four different algorithms including Alpha miner, Heuristic miner, ILP miner and Inductive miner to generate process models that illustrate students learning behavior. Conformance checking was done using different quality metrics in order to evaluate performance. These metrics involves simplicity, generalization, F1 score, precision and fitness.

Sequential pattern mining is used for interaction pattern mining. We use Generalized Sequential Pattern (GSP) algorithm for this purpose and cluster students on the basis of similar patterns using K-means clustering technique.

Finally we represent their performance based on their interactions using normal distribution and validate results. Figure 3-1 shows our proposed framework.

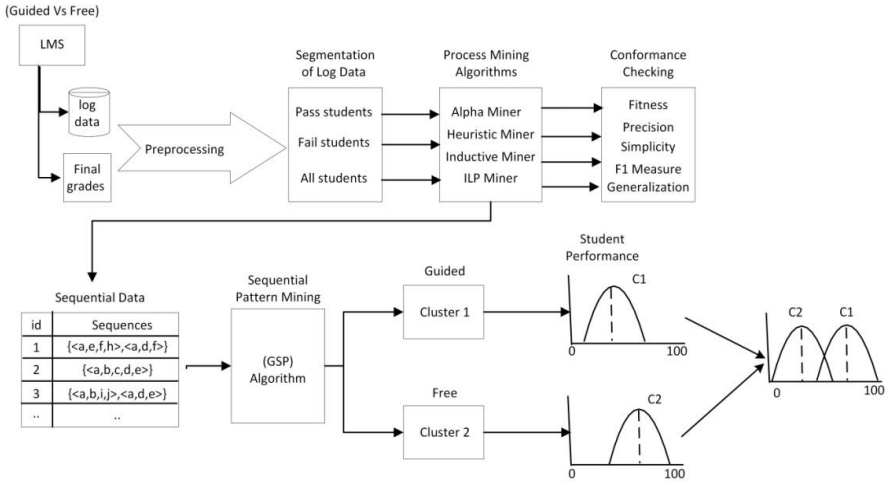


Figure 3-1 Proposed framework

3.3 System Design:

For research purpose, we develop Learning Management System (LMS) with which student can interact and access learning content along with numerous features as follows:

- Watch video lectures.
- Download handouts.
- View quiz list.
- Attempt quizzes.
- Download or upload assignments.
- View progress.
- View notifications.
- View announcements.
- View events.
- Use Discussion forum.
- Communicate with teacher and other class fellows.

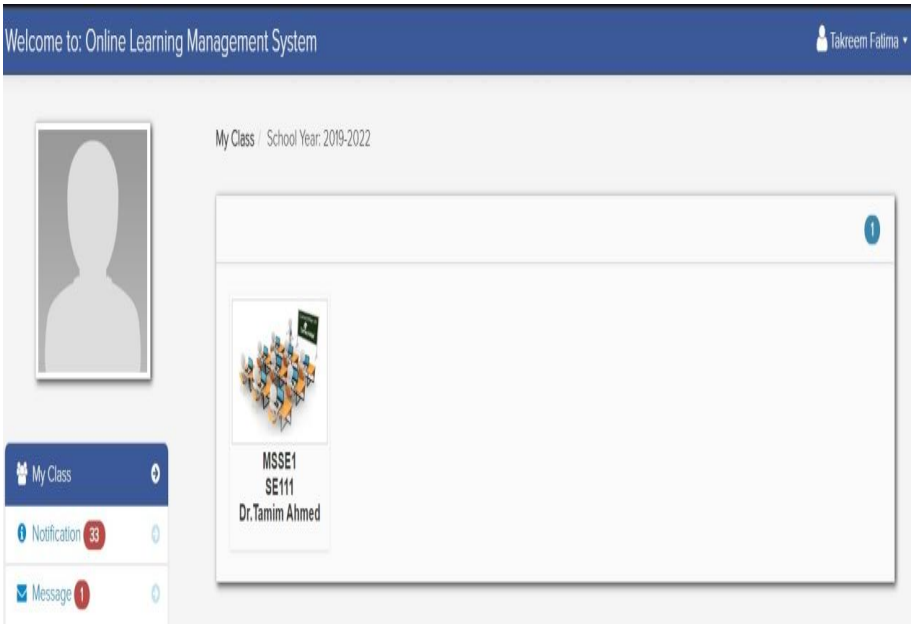
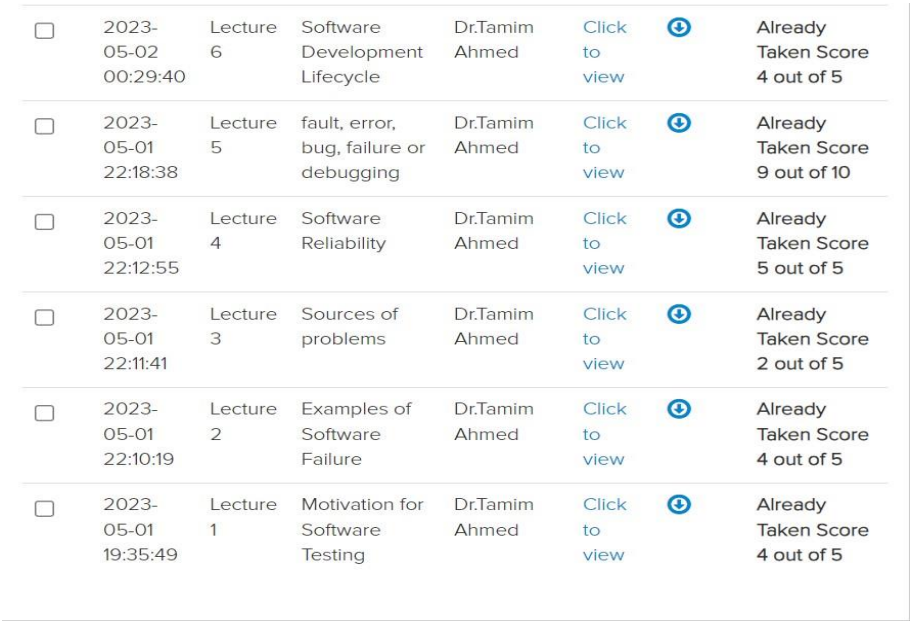


Figure 3-2 System Interface

Figure 3-3 LMS features

In conjunction with the functionalities provided by the e-learning platform LMS depicted in Figure 3-2 and Figure 3-3, the design of e-learning courses holds a significant importance, encompassing the arrangement and structure of both content and activities. Typically, instructors take charge of organizing the course sequence, incorporating content, activities, and avenues for communication, aligning them with their didactic and pedagogical strategies to effectively guide students in their learning journey throughout the course. Nonetheless, unless specific limitations are imposed, LMS platforms offer students the flexibility to opt for alternative paths divergent from the predefined course structure and organization [26].



☐	2023-05-02 00:29:40	Lecture 6	Software Development Lifecycle	Dr.Tamim Ahmed	Click to view		Already Taken Score 4 out of 5
☐	2023-05-01 22:18:38	Lecture 5	fault, error, bug, failure or debugging	Dr.Tamim Ahmed	Click to view		Already Taken Score 9 out of 10
☐	2023-05-01 22:12:55	Lecture 4	Software Reliability	Dr.Tamim Ahmed	Click to view		Already Taken Score 5 out of 5
☐	2023-05-01 22:11:41	Lecture 3	Sources of problems	Dr.Tamim Ahmed	Click to view		Already Taken Score 2 out of 5
☐	2023-05-01 22:10:19	Lecture 2	Examples of Software Failure	Dr.Tamim Ahmed	Click to view		Already Taken Score 4 out of 5
☐	2023-05-01 19:35:49	Lecture 1	Motivation for Software Testing	Dr.Tamim Ahmed	Click to view		Already Taken Score 4 out of 5

Figure 3-4 Pre-defined sequence in LMS

We design LMS in a way that students can either access the content in guided or in a free manner. As seen in above Figure 3-4Figure 3-3 we set predefined sequence to (watch video lectures, download handouts, attempt quiz) in order to see the impact of guided vs free use of LMS on student performance. For analysis of student performance

based on their interactions, we record student interactions in a log file and also keep track of their performance.

The LMS is developed using PHP and other web languages like HTML 5, CSS 3 , Bootstrap 6, AJAX, JSON, Javascript, on local host XAMP server database was maintained using SQL on Phpmyadmin. Later we live it on server, so every registered student can access and use features available on it.

3.4 Data Collection

This study utilizes data obtained from a virtual university's undergraduate-level "Software Verification and Validation" course. The course spanned 6 weeks and was conducted through an e-learning platform, specifically the Learning Management System (LMS). A total of 76 students actively participated in the course, engaging in various activities and receiving feedback for their submissions. The students were required to watch video lectures, study relevant texts, and attempt quizzes associated to each lecture uploaded on LMS on certain topic, quizzes were automatically evaluated. The assessment process involved assigning grades to each quiz and calculating the overall percentage achieved across all quizzes, which served as the final grade for each student. Initial criteria or threshold value of passing is greater than 33% in each quiz. Reward system was associated with student performance where rating represents their performance in each quiz. This was helpful in visualize performance and motivate them to perform better. Overall in final grade we consider students having their final grades greater than 60 are high achievers and those who scored below 60 are consider as low performers. Subsequently, the students' "log" files and "final grades" were exported in CSV format to facilitate analysis as depicted in Figure 3-5.

1	User_id	Grades
2	220	100
3	221	68
4	222	34
5	223	83
6	224	37
7	225	77
8	226	85
9	227	85
10	228	74
11	229	80
12	230	34
13	231	25
14	232	77
15	233	28
16	234	82
17	235	31
18	236	28
19	237	68
20	238	88
21	239	22
22	240	100
23	241	37

Figure 3-5 Final grades csv

3.4.1 Log Data

In the field of process mining, event data or LMS usage data serves as the primary data source, commonly known as an event log. These event logs are automatically gathered by the learning management system (LMS), capturing the navigation actions of students. Every time a student clicks on a hyperlink, this action is automatically recorded in the event logs as illustrated in Figure 3-6 and Figure 3-7. Each event corresponds to an activity executed by a specific resource within a particular time and case. Additional attributes may be included in these events to provide supplementary information. Event logs comprise a collection of traces, as it represents the chronological order of activities or events pertaining to a specific case. Event logs essentially keep track of all the actions taken while a process is running. Each time the procedure is used, a case is created, and each case creates a trace—a series of activity occurrences. These occurrences have many qualities and are recorded as events [79].

1	log_id	User_id	Username	Date	Actions
2	7128	223	Hafiz Muhammad	01/01/23 11:34:38am	user Loged in
3	7129	223	.Hafiz Muhammad.	01/01/23 11:34:39am	view notifications
4	7130	223	.Hafiz Muhammad.	01/01/23 11:36:09am	view lectures
5	7131	223	.Hafiz Muhammad.	01/01/23 11:37:25am	.viewed lecture link https://www.youtube.com/watch?v=vO7QXLTSE4I.
6	7132	223	.Hafiz Muhammad.	01/01/23 11:38:42am	.downloaded lecture file Lecture 1.
7	7133	223	.Hafiz Muhammad.	01/01/23 11:39:59am	attempt Quiz
8	7135	223	.Hafiz Muhammad.	01/01/23 11:40:30am	view progress
9	7136	223	.Hafiz Muhammad.	01/01/23 11:41:49am	user logged out
10	7137	220	Hira	01/01/23 11:45:03am	user Loged in
11	7138	220	.Hira.	01/01/23 11:45:04am	view notifications
12	7139	220	.Hira.	01/01/23 11:46:48am	view lectures
13	7140	220	.Hira.	01/01/23 11:48:02am	.viewed lecture link https://www.youtube.com/watch?v=vO7QXLTSE4I.
14	7141	220	.Hira.	01/01/23 11:49:18am	.downloaded lecture file Lecture 1.
15	7142	220	.Hira.	01/01/23 11:49:46am	attempt Quiz
16	7144	220	.Hira.	01/01/23 11:50:25am	view progress
17	7145	220	.Hira.	01/01/23 11:51:45am	user logged out
18	7146	241	Zar	01/01/23 11:54:05am	user Loged in
19	7147	241	.Zar.	01/01/23 11:54:06am	view notifications
20	7148	241	.Zar.	01/01/23 11:55:23am	view lectures
21	7149	241	.Zar.	01/01/23 11:56:40am	.viewed lecture link https://www.youtube.com/watch?v=vO7QXLTSE4I.
22	7150	241	.Zar.	01/01/23 11:58:05am	view assignment
23	7151	241	.Zar.	01/01/23 11:59:17am	user logged out
24	7152	237	Haroon	01/01/23 12:00:31pm	user Loged in

Figure 3-6 Student logs csv 1

342	7507	277	Nayab	01/02/23 11:24:53pm	user Loged in
343	7508	277	.Nayab.	01/02/23 11:24:54pm	view notifications
344	7509	277	.Nayab.	01/02/23 11:26:19pm	view Quiz List
345	7510	277	.Nayab.	01/02/23 11:27:29pm	attempt Quiz
346	7512	277	.Nayab.	01/02/23 11:27:54pm	view progress
347	7513	277	.Nayab.	01/02/23 11:29:01pm	view assignment
348	7514	277	.Nayab.	01/02/23 11:30:10pm	view lectures
349	7515	277	.Nayab.	01/02/23 11:31:23pm	.downloaded lecture file Lecture 1.
350	7516	277	.Nayab.	01/02/23 11:32:33pm	user logged out
351	7538	224	Madiha	01/03/23 09:36:09pm	user Loged in
352	7539	224	.Madiha .	01/03/23 09:36:10pm	view notifications
353	7540	224	.Madiha .	01/03/23 09:36:17pm	view Quiz List
354	7541	224	.Madiha .	01/03/23 09:38:20pm	attempt Quiz
355	7543	224	.Madiha .	01/03/23 09:40:35pm	view progress
356	7544	224	.Madiha .	01/03/23 09:42:00pm	view assignment
357	7545	224	.Madiha .	01/03/23 09:43:07pm	view lectures
358	7546	224	.Madiha .	01/03/23 09:44:19pm	.viewed lecture link https://www.youtube.com/watch?v=vO7QXLTSE4I.
359	7547	224	.Madiha .	01/03/23 09:45:30pm	user logged out
360	7548	227	Muhammad Umer	01/03/23 09:47:16pm	user Loged in
361	7549	227	.Muhammad Umer	01/03/23 09:47:17pm	view notifications
362	7550	227	.Muhammad Umer	01/03/23 09:48:35pm	view lectures
363	7551	227	.Muhammad Umer	01/03/23 09:49:45pm	.viewed lecture link https://www.youtube.com/watch?v=vO7QXLTSE4I.
364	7552	227	.Muhammad Umer	01/03/23 09:51:52pm	.downloaded lecture file Lecture 1.
365	7553	227	.Muhammad Umer	01/03/23 09:54:59pm	attempt Quiz

Figure 3-7 Student logs csv 2

The log file acquired from the Learning Management System (LMS) initially contained 37405 event logs characterized by five attributes as listed below in Table 3-1**.**

Table 3-1 Attributes of student logs csv

Attributes	**Description**
Log id	Unique Id of each record
User id	Unique id assign to students
User name	Name of registered students
Date	Contains timestamp when action performed
Action	Activity performed by students on LMS

3.5 Data Pre-processing

The renowned idiomatic phrase "garbage in, garbage out" holds great relevance in the context of Process Mining (PM), as the attainment of meaningful outcomes heavily relies on the utilization of high-quality event logs that accurately capture the execution of all pertinent variations within a given business process [80]. We performed following preprocessing steps on collected data before applying process discovery algorithms.

3.5.1 Data Cleaning

Before data cleaning students performed actions while navigating through learning management system (LMS) as shown in Table 3-2**.**

Table 3-2 Student activities on LMS before data cleaning

#	**Activities**	**Description**
1	User logged in	Registered student logged in to LMS
2	View notifications	Student view notification of lectures, assignment and quiz uploaded by teacher.
3	View lecture	Student view lectures page
4	Watch video lecture	Students click on hyperlink to watch video lecture
5	Download handouts	Student download lecture notes
6	Attempt Quiz	Student take quiz
7	View progress	Student view progress page
8	View Quiz list	Students view quizzes
9	View assignments	Student view assignment page
10	Download assignment	Student download assignments uploaded by teacher
11	Upload Solution	Students submit assignment solution
12	View Events	Student view class calendar and update themselves regarding upcoming class events
13	View CLO	Students view course overview and course learning outcomes
14	View student list	Student view their classmates
15	View messages	Student view messages
16	User Logout	Student logged out

In data cleaning step we remove log events under student actions that are irrelevant in context of analyzing their performance. We remove view notification; view CLO, view calendar, view events, and view student list actions as we previously explained in Table, performing these activities have no impact on student grades. Such features are available on LMS just for student information. Table 3-3 shows actions related to student performance.

Table 3-3 Student actions on LMS after data cleanining

#	**Relevant Actions**
1	User logged in
2	View lectures
3	Watch video lectures
4	Download handouts
5	Attempt quiz
6	View progress
7	View assignments
8	View Quiz list
9	View announcements
10	User Logged out

After data cleaning, when we considered only relevant actions of students as depicted in Table, our log data reduced to 37405. This cleaned data is considered for further preprocessing.

3.5.2 Noise Removal

For precise and accurate process model we deal with noisy data including handling missing values and removing duplicate records or data, such noisy data can affect model efficiency and accuracy.

3.5.3 Symbolization of operations

We symbolize student actions with alphabets that are helpful in processing and representation of activities and also in creating sequential dataset. We symbolize activities as depicted in Table 3-4**.**

Table 3-4 Symbolization of operations

Symbolization	**Activities**
A	User login
B	View lectures
C	Watch video lectures
D	Download handouts
E	Attempt quiz
F	View quiz list
G	View progress
H	View assignment
I	View announcement
J	User logout

3.5.4 Data Transformation

PM4Py offers conversion utilities that facilitate the transformation of event data objects between different formats. Additionally, PM4Py provides support for pandas data frames, which prove to be highly efficient when dealing with extensive event data sets [79].

We transform CSV data into the XES format, which is the default input file format used by process mining discovery techniques. We have done this conversion using

PM4PY conversion utilities that allow us to convert log data into specified format for processing.

3.5.5 Data Integration

We merge student logs with final grades of students at common attribute Case_id of the event log.

3.5.6 Segmentation of Log Data

[80] Define segmentation as ‘splitting event log into different clusters or group of cases’. In the context of a collection comprising 'n' instances, a segment denoted as 's' embodies the amalgamation of these instances, expressed as follows: $s = (c1 \cup c2 \cup,...,cn)$, where 's' represents a subset inherently contained within an Event Log, symbolically represented as $S \subseteq L$. Clustering was used as a pre-processing step in the previous studies, since learning is a difficult process and they wanted to make it better and simpler. In this regard, numerous studies [26] [47] [49] opted for a manual clustering approach, wherein students were grouped solely based on their final grades.

Three separate student clusters appeared after using K-means clustering algorithms in this study to group students based on their final grades:

- **Cluster 1:** Consisted of 48 students who achieved grades of 60 or above.
- **Cluster 2:** Encompassed a group of underperforming students whose grades fell below 60.
- **Cluster 3:** Comprised of all students who participated in the online course.

3.6 PROM Tool

We used PROM framework tool for statistical analysis of log data. Figure 3-8 displays the user interface of the PROM tool, comprising three primary features positioned at the top of the application: the Object View, Action View, and Visualization View. Additionally, an integral function offered by the tool is the import button, designed to facilitate in importing log files ino the tool.

Figure 3-8 PROM Framework user interface

In order to make the files compatible with the ProM tool (Van der Aalst, 2011a), a necessary step was taken to convert them into the appropriate format. Initially, the Moodle log file was saved in comma-separated values (CSV) format, as depicted inFigure 3-6 and Figure 3-7. Subsequently, the CSV file underwent conversion into mining extensible markup language (MXML), which is the format that ProM interprets. MXML is an XML-based format designed for the interchange of event logs. It emerged in 2003 as the pioneering standard, gaining adoption by the PROM (Process Mining Manifesto)

initiative. MXML sets forth a standardized notation for the storage of dates, resources, and transaction types.

However, in 2010, XES took over from MXML as the new process mining format, operating independently of any specific tool. XES draws on the practical insights gleaned from MXML but offers a less restrictive and more genuinely extensible framework.

For this conversion process, we utilized the PROM Import Framework .To accomplish the conversion successfully, we specifically opted for the "General CSV File" option from the "Filter" properties tab, as shown in Figure 3-9**.**

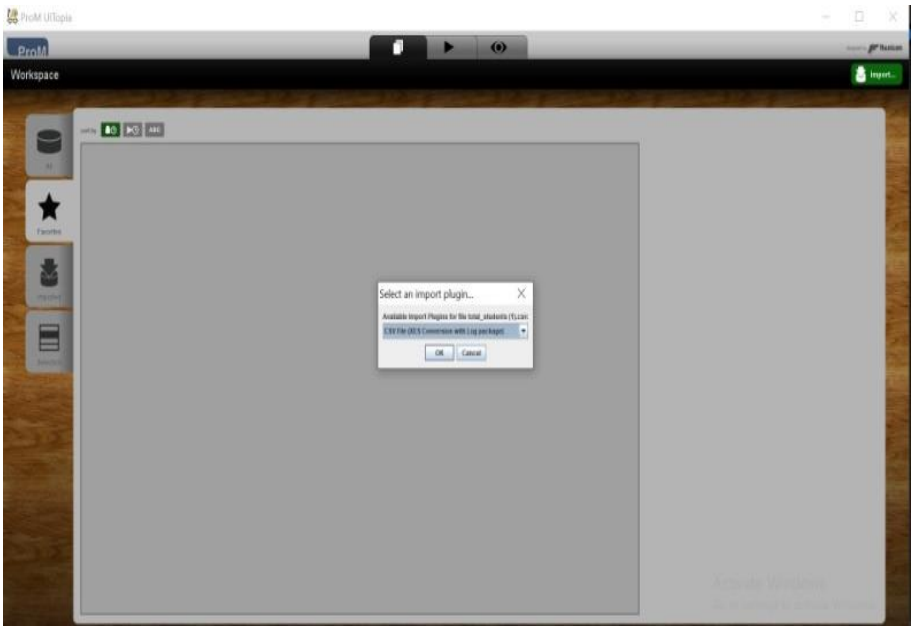


Figure 3-9 Import csv log file

Additionally, we ensured the proper linkage between the names in the head of the CSV file and their corresponding labels in the properties panel as shown in Figure 3-10.

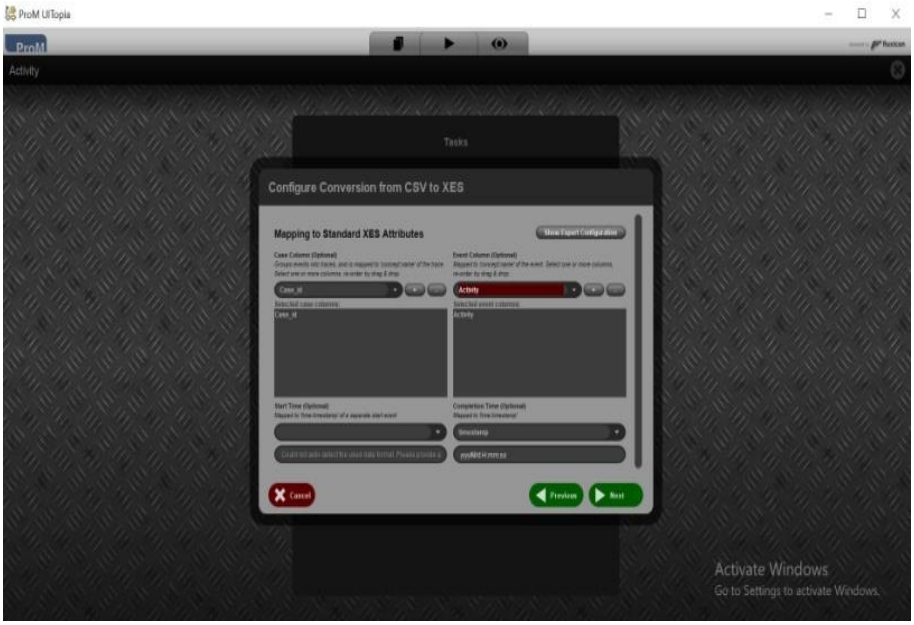


Figure 3-10 CSV to XES conversion process

- The property denoted as "Case ID" was associated with the corresponding value of "Case id"
- The property known as "Task ID" was connected to the value representing "Activity"
- The property labeled as "Start Time" was linked to the value indicating "Timestamp"

Furthermore, it is most important to ensure the accurate configuration of the "Date Format" field, which should be according to the format of "Day-Month-Year-Hour:Minute."

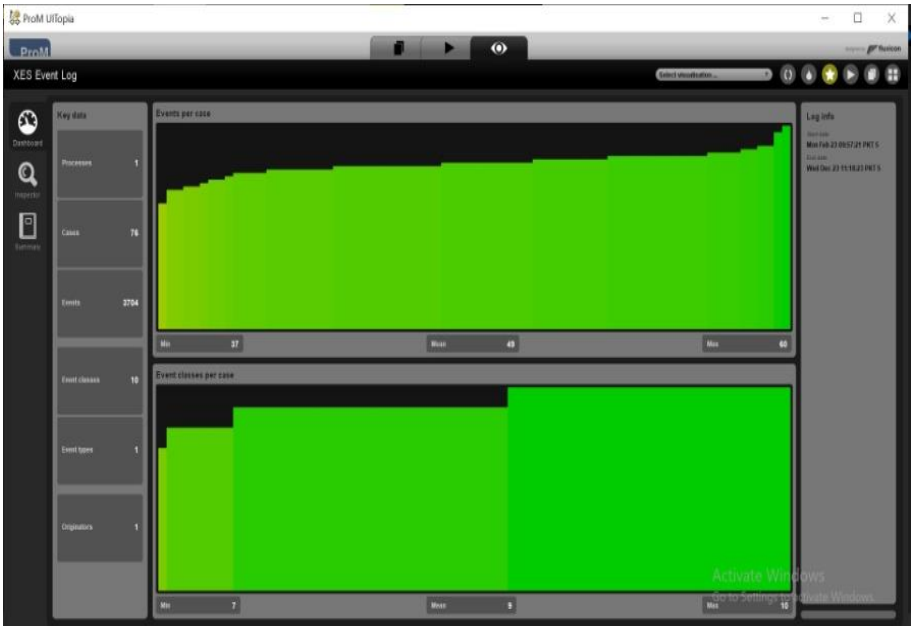


Figure 3-11 Student logs statistics in PROM

Figure 3-11 illustrates the quantitative analysis conducted on a log file, illustrating statistical information pertaining to the quantity of procedures, occurrences (examples, sequences), incidents, categories of occurrences, types of events, and initiators, (depicted in the leftmost column). Within the context of individual cases, the metric "Events per Case" delineates the minimum, mean, and maximum count of events, while an analogous representation is observed for "Event Classes per Case."

3.7 PM4PY:

Process Mining for Python library (PM4Py) aims to bridge the gap between process mining and data science. The library integrates with other data science libraries such as pandas, numpy, scipy, and scikit-learn, and offers algorithmic customization as well as support for process discovery, conformance checking, and process enhancement [79].

Alpha miner, heuristic miner, ILP miner, also inductive miner process discovery algorithms are implemented in our work utilizing PM4PY in the Python programming language. Visualization of process models and its petrinets also generated through same approach.

We also perform conformance checking using PM4py library in python for measure deviation of modeled and actual behavior of process model and to measure model efficiency and accuracy using various conformance checking techniques like simplicity, precision, generalization, F1 score fitness and accuracy. These all approaches aids in comparing algorithm accuracy and helps us to select best algorithm overall.

3.8 Application of Process Mining Algorithms

Process models are extracted from event logs using process discovery algorithms. The Alpha Miner, Inductive Miner, ILP Miner, and Heuristic Miner are just a few algorithms that employ various strategies to build process models.

3.8.1 Alpha miner:

Let L be an event log over T. α(L) is defined in Figure 3-12 as explained by [81].

1) $T_L = \{t \in T \mid \exists_{\sigma \in L}\, t \in \sigma\}$,
2) $T_I = \{t \in T \mid \exists_{\sigma \in L}\, t = first(\sigma)\}$,
3) $T_O = \{t \in T \mid \exists_{\sigma \in L}\, t = last(\sigma)\}$,
4) $X_L = \{(A,B) \mid A \subseteq T_L \wedge A \neq \emptyset \wedge B \subseteq T_L \wedge B \neq \emptyset \wedge \forall_{a \in A}\, \forall_{b \in B}\, a \rightarrow_L b \wedge \forall_{a_1, a_2 \in A}\, a_1 \#_L a_2 \wedge \forall_{b_1, b_2, \in B}\, b_1 \#_L b_2\}$,
5) $Y_L = \{(A,B) \in X_L \mid \forall_{(A',B') \in X_L} A \subseteq A' \wedge B \subseteq B' \Rightarrow (A,B) = (A',\ B')\}$,
6) $P_L = \{p_{(A,\ B)} \mid (A,\ B) \in Y_L\} \cup \{i_L,\ o_L\}$,
7) $F_L = \{(a, p_{(A,B)}) \mid (A,B) \in Y_L \wedge a \in A\} \cup \{(p_{(A,B)},\ b) \mid (A,B) \in Y_L \wedge b \in B\} \cup \{(i_L, t) \mid t \in T_I\} \cup \{(t, o_L) \mid t \in T_0\}$, and
8) $\alpha(L) = (P_L,\ T_L,\ F_L)$.

Figure 3-12 Alpha miner algorithm

The α-algorithm takes an event log (L) consisting of a set or sequence of activities (<a1, a2… an>) as its input [35] . This algorithm comprises eight steps:

1. Each entry within the event log (referred to as TL) is representative of a specific transition within a workflow net (WF-net).
2. Determine the grouping of initial actions, where "TI" signifies the foundational components of a given sequence.
3. Recognize the collection of concluding tasks, denoted as "TO," which symbolizes the ultimate components within any given sequence.
4. Discover both events, indicated as (A, B), where every component in A and each component in B are directly associated, i.e., (a, b) € A×B: $a \rightarrow_L b$, and input transitions in A are represented as ($\bullet P_{(A,B)} = A$) where as B consists of set of output transitions ($P_{(A,B)} \bullet = B$). Furthermore, elements in A should never come after any other components, i.e., a_1, a_2 € A: $a_1 \#_L a_2$. The same rule also applies to B.

5. Remove any previously found (A, B) pairs that are not maximal.
6. Assign the label P(A,B) to each pair (A, B) within the Petri Net framework. Introduce an initial position denoted as iL, along with a concluding position labelled as oL.
7. Draw an arc from every place P(A,B) to each element in its set A of source transitions and set B of target transitions. Include arcs from the source place (TI) to each start transition (iL) and from each end transition (To) to the final place (oL).
8. The final model $\alpha(L)= (P_L, T_L, F_L)$ consists of all the defined places, transitions, and arcs.

3.8.2 Heuristic Miner

This method enhances the alpha algorithm by incorporating trace frequency analysis within the log data. The control flow aspects of a process model are mined by the Heuristics Miner by exclusively examining event sequencing within individual cases. In practical terms, this entails focusing solely on the case ID, timestamp, and activity fields within the log file during the mining process. The chronological arrangement of activities is determined through the utilization of activity timestamps.

The process involves a sequence of three distinct stages:

1. Developing the graph of dependency.
2. Crafting Formulating input and output expressions for each distinct activity.
3. Explore long distant dependency relationships.

Let W denote an event log pertaining to the set T, that is, W is a member of T. Let a and b be elements of set T.

1. The relation a >W b is valid exclusively when a trace exists S = t1, t2, t3,…, tn and an index i belonging to the set {1, 2, ..., n - 1}, such that S is present in W, ti equals a, and ti+1 equals b.
2. The implication a →W b is satisfied only when a >W b and b >W a are both met.
3. The statement a ≠ W b is valid when a >W b and b >W a are both negated.
4. The condition a ‖ W b is fulfilled only if a >W b and b >W a are concurrently met.
5. The relation a>>W b is established only when there exists a sequence S = t1, t2, t3, ..., tn and an index i belonging to the set {1, 2, ..., n - 2}, such that S is part of W, ti equals a, ti+1 equals b, and ti + 2 equals a.
6. The scenario a >>>W b holds true in case if there exists a path S = t1, t2, t3, ..., tn and indices i and j, where i is less than j and both i and j belong to the set {1, 2, ..., n}, such that S is present in W, ti equals a, and tj equals b.

Upon deducing the correlation based on occurrence frequencies, our process commences with the establishment of what is commonly referred to as a dependency graph. A metric grounded in frequency analysis is employed to signify the level of confidence regarding the existence of a dependency link between two occurrences labeled as 'a' and 'b' (denoted as 'a W b'). The ascertained 'W' values among events within a given event log are then harnessed within a heuristic exploration aimed at determining accurate dependency relationships [82].

$$a \rightarrow w\, a = \left(\frac{|a>wb|-|b>wa|}{|a>wb|+|b>wa|+1}\right) \qquad \textit{Equation 1}$$

The value of Wb resides within the interval of -1 to 1.

In the context of brief sequences, the interdependence is quantified in the subsequent manner: Consider an event log denoted as W spanning across a timeframe T,

wherein a and b are elements of T. Consequently, |a >W a| signifies the frequency of occurrences of a >W a within W, whereas |a >>W b| corresponds to the frequency of instances where a >>W b occurs within the event log W.

$$a \rightarrow_W a = \left(\frac{|a >_W a|}{|a >_W a| + 1}\right) \qquad \textit{Equation 2}$$

$$a \rightarrow_{2W} b = \left(\frac{|a \gg_W b| - |b \gg_W a|}{|a \gg_W b| + |b \gg_W a| + 1}\right) \qquad \textit{Equation 3}$$

The event log W is characterized as a multiset, wherein identical traces have the potential to manifest multiple times within the log, and patterns can recur multiple times within a single trace [82].

3.8.3 ILP Miner Algorithm

Event log is considered, denoted as L, which encompasses the event collection, referred to as AL, and associated matrices M, M′, and ML, the subsequent exposition pertains to real-valued variable cm within the domain R i.e., cm ∈ R, alongside ~cx and ~cy, also real-valued variables existing within the range of R and aligned with the cardinality of activity set AL i.e., cx, ~cy ∈ $R^{|AL|}$. The crux of the matter lies in the formulation of the ILP (Integer Linear Programming) dedicated to process discovery, abbreviated as ILP_L. This formulation encapsulates the essence of deriving process insights from the provided event log and its associated matrices. Basic Formulation of ILP miner algorithm is defined in Figure 3-13.

$$
\begin{array}{lrl}
\textbf{\textit{minimize}} & z = c_m m + \vec{c_x}^{\intercal}\vec{x} + \vec{c_y}^{\intercal}\vec{y} & \textit{objective function} \\
\textbf{\textit{such that}} & m\vec{1} + \mathbf{M}'\vec{x} - \mathbf{M}\vec{y} \geq \vec{0} & \textit{theory of regions} \\
\textbf{\textit{and}} & m\vec{1} + \mathbf{M}_L(\vec{x} - \vec{y}) = \vec{0} & \textit{corresp. place is empty after each trace} \\
 & \vec{1}^{\intercal}\vec{x} + \vec{1}^{\intercal}\vec{y} \geq 1 & \textit{at least one arc connected} \\
 & \vec{0} \leq \vec{x} \leq \vec{1} & \textit{i.e. } \vec{x} \in \{0,1\}^{|A|} \\
 & \vec{0} \leq \vec{y} \leq \vec{1} & \textit{i.e. } \vec{y} \in \{0,1\}^{|A|} \\
 & 0 \leq m \leq 1 & \textit{i.e. } m \in \{0,1\}
\end{array}
$$

Figure 3-13 Basic formulation of ILP miner algorithm

3.9 Process Visualization

We use Graphviz library with PM4PY for the representation and visualization of process trees, process models, directly follow graphs and petrinets.

3.9.1 Petrinets

Petri net is a tuple (P, T, F) where:

1. P is a finite set of places,
2. T is a finite set of transitions such that P Π T = φ and
3. 'F ⊆ (P × T) ⊔ (T × P)' constitutes a collection of directed arcs recognized as the flow relation.

3.10 Conformance Checking:

We perform conformance checking of process models obtained from the application of process discovery algorithms of process mining, We used following standard quality

metrics to measure outcomes of process models. These quality metrics includes Generalization, Simplicity, Fitness, Precision and F1 measure as explained in detail in below section.

3.10.1 Generalization

A model is said general when all of its nodes receive a sufficient number of visits during the replay of a log on the said model. We can compute generalization of a specified model using the following formula explained in Figure 3-14**.**

$$g = 1 - avg_t\left(\sqrt{\frac{1}{freq(t)}}\right)$$

Figure 3-14 Generalization formula

In above equation, average of overall log transition is represented by avg t, whereas after the replay the frequency of transition is represented by freq t.

3.10.2 Precision

The precision metric is calculated by testing the model's ability to predict the next activity in the process based on a log of previous activities. The set of possible transitions that follow an activity in the process model is compared to the activities that follow the same prefix in the log. A higher number of differences between the two sets indicates a lower precision score, and vice versa. Within the PM4PY framework, two distinct methodologies are incorporated a token-based approach and an alignment methodology [83]. Equation presents the precision formula (p,Ɛ), in which represents obs p(e) the observed behavior, and pos p(e) denotes the potential behavior [84].

$$Precision\ (p,\mathcal{E}) = \frac{\sum_{e \in \mathcal{E}} \{obsp(e)\}}{\sum_{e \in \mathcal{E}} \{posp(e)\}} \qquad Equation\ 4$$

3.10.3 Fitness:

Let us consider 'k' as various sequences derived from an event log. Let 'i' represent the sequence number, where i ranges from 1 to k ($1 \leq i \leq k$). Now, we define certain variables related to these sequences: 'pi' denotes the tokens produced during the log replay, 'ci' refers to the tokens consumed, 'ri' represents the tokens that remained, and 'mi' indicates the missing tokens. It is important to note that for any given 'i', the value of 'mi' is less than or equal to 'ci', 'ri' is less than or equal to 'pi', and 'f' lies between 0 and 1 i.e., ($i, mi \leq ci , ri \leq pi$) and ($0 \leq f \leq 1$) [69].

$$F = \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i \ m_i}{\sum_{i=1}^{k} n_i \ m_i}\right) + \frac{1}{2}\left(1 - \frac{\sum_{i=1}^{k} n_i \ \gamma_i}{\sum_{i=1}^{k} n_i \ \rho_i}\right) \qquad Equation\ 5$$

3.10.4 F1 Score

The F1 score serves as a performance metric for evaluating the accuracy of a process model. It is commonly computed as the harmonic mean of both precision and recall. This metric assesses the accuracy of the process model in capturing positive event traces as well as negative ones that do not conform to the model. Thus, when the process model categorizes all traces as positive, the F1 score reaches its maximum value of 1 (100%). Conversely, if all traces are classified as negative, the F1 score reaches its minimum value of 0 (0%). In general, the F1 score spans a range between 0% and 100%, providing an inclusive evaluation of model accuracy [33].

F1-Score = ‘ 2 ∗ (precision ∗ recall) / (precision + recall) ’ *Equation 6*

3.10.5 Simplicity

The concept of simplicity is rooted in the measurement of arc degrees, which signifies the average count of both incoming and outgoing arcs computed for each node. As outlined by Blum (2015), this concept involves assessing the contrast between the weighted mean arc degree within the derived (indexed as "m") and the initial (indexed as "o") models. A greater contrast signifies a more complex model, consequently resulting in a lower score [83]. Equation (4) presents the simplicity formula S (L, M_m, M_o), wherein $S`_M$ represents the model M's weighted average arc degree [84].

$$S\ (L, M_m, M_o) = \frac{1}{1+max\ \{O, S`Mo - S`Mm\}} \qquad \textit{Equation 7}$$

3.11 Application of Sequential Pattern Mining

Pattern mining is an automated process that uncovers concealed patterns within data, with the primary aim of identifying patterns that can be easily understood by humans. In this context, we focus on widely recognized technique for sequential pattern mining (SPM) (Fournier-Viger et al., 2017).

3.11.1 Generalized Sequential Pattern

The GSP algorithm, proposed in [85], performs a similar function to the AprioriAll algorithm but eliminates the need to initially identify all frequent itemsets. This algorithm offers several advantages in terms of pattern analysis:

- It allows for the imposition of time constraints on the temporal separation between consecutive elements within a pattern.
- It permits the inclusion of items within a pattern element that span a transaction set within a user-specified time window.
- It facilitates pattern discovery at various levels of a user-defined taxonomy.

Moreover, GSP is specifically designed for the detection of generalized sequential patterns. The GSP algorithm operates through multiple passes over a sequence database, following this process:

- During the first pass, it identifies the frequent sequences that meet the minimum support requirement.
- At each subsequent pass, the algorithm examines each data sequence to update the occurrence count of the candidates contained within that sequence.

Figure 3-15 depicted the pseudo code of GSP algorithm as explained in [45]

- ✓ Obtain a sequences in form of <x> as length-1 candidates
- ✓ find F_1 (the set of length-1 sequential patterns), after a unique scan of database
- ✓ Let k=1;

While F_k is not empty do

- Form C_{k+1}, the set of length-(k+1) candidates from F_k;
- If C_{k+1} is not empty, unique database scan, find F_{k+1} (the set of length-(k+1) sequential patterns)
- Let k=k+1;

End While

Figure 3-15 GSP algorithm pseudo code

The GSP algorithm necessitates temporal and case ID attributes. These paired attributes serve the purpose of extracting a chronological sequence for each case, notably, the students under examination. This sequence encompasses all events or activities arranged chronologically. GSP methodically probes for commonly recurring patterns within these events, subsequently unveiling the most pervasive ones for rule formulation.

Operational within the GSP algorithm is a defined set of parameters, including but not limited to, minimal support, minimum gap, maximum gap, and window size. The parameter of minimal support designates the proportion of cases required to classify a pattern as frequent. The window size parameter determines the temporal extent during which a successive activity or event is categorized under the same case. This parameter holds diminished significance within this study, as any distinct event or activity is treated as a discrete state of the same case. Furthermore, the parameter of maximum gap regulates the inclusion of sequences wherein pattern occurrences are temporally distant. Similarly, the "minimum gap" parameter undertakes a parallel role when activities exhibit close temporal proximity [78].

CHAPTER 4

RESULTS AND EVALUATION

4.1. Alpha Miner

Figure 4-1 and Figure 4-3 shows the process models generated from alpha miner algorithm for the process discovery of event logs extracted from LMS, which records the students navigation paths that depict their interactions with LMS and their learning behavior, We observed from following process models obtained from alpha miner that in Figure 4-3 that illustrate the navigation paths of low performer students have more diverse behavior than those in high performer group in Figure 4-1**.**

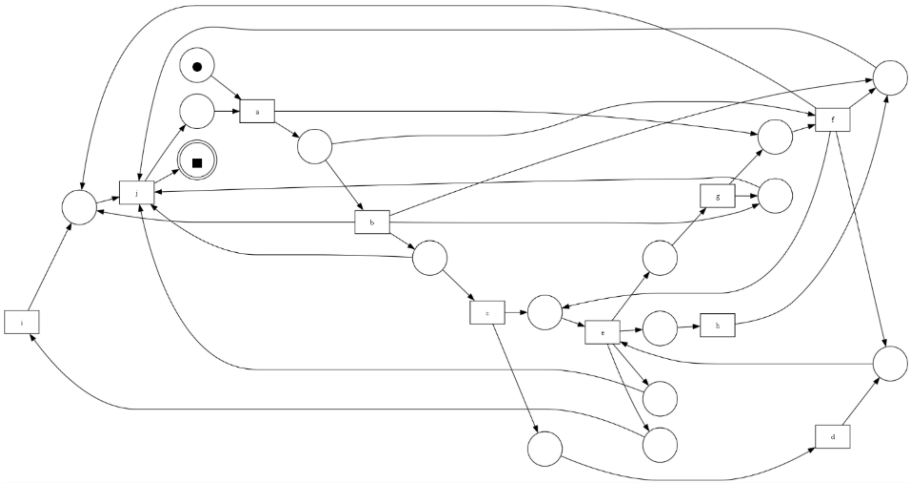


Figure 4-1 High performers Process model obtained using Alpha miner

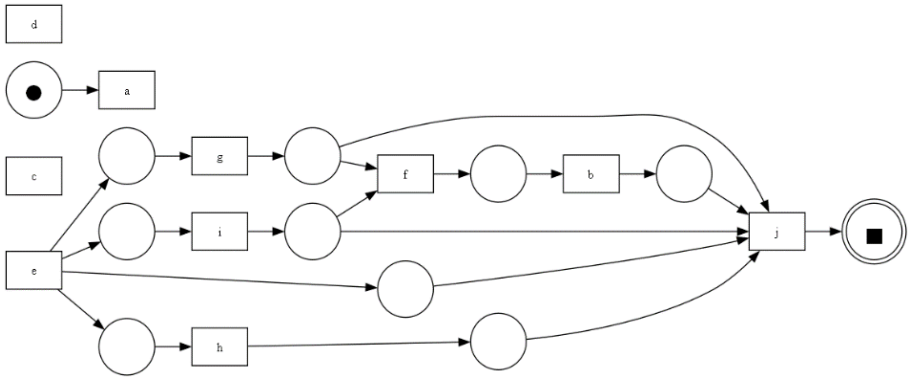


Figure 4-2 High performers Petrinet obtained using Alpha miner

In Figure 4-1 it is observed that students follows predefined path as initially set by instructor and have less deviations in their learning path, but the students in Figure 4-3 adopted more diverse navigational paths and deviated from predefined path and explore LMS in free manner. Petrinet diagrams in Figure 4-2 and Figure 4-4 shows the behavior of process models obtained from the implementation of alpha miner, These representations illustrates in detail the flow of events, deviations and bottleneck analysis of these models.

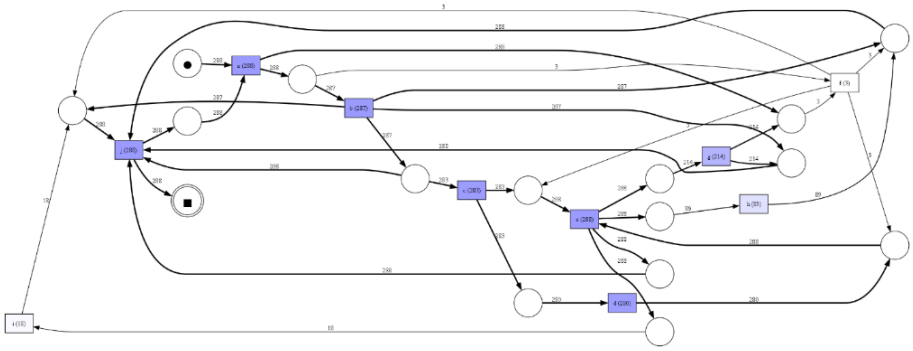


Figure 4-3 Low performers Process model obtained using Alpha miner

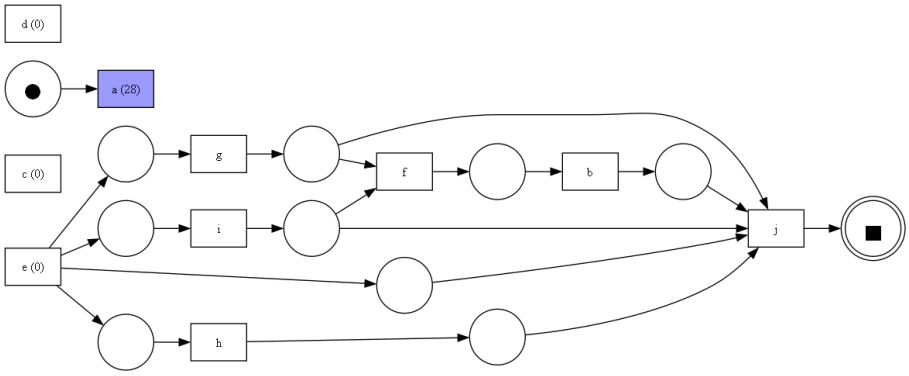


Figure 4-4 Low performers Petrinet obtained using Alpha miner

4.2 Heuristic Miner:

The Heuristics Miner algorithm, as introduced by Van der Aalst in 2011, presents a heuristic network in the form of a cyclic, directed graph that depicts the prevalent patterns of student behavior while navigating through a course in LMS. In this visual depiction, the square-shaped containers symbolize the activities executed by students during their interaction with the Learning Management System (LMS) interface. The curved lines, or connections, illustrate the interdependencies and associations that exist among these activities.

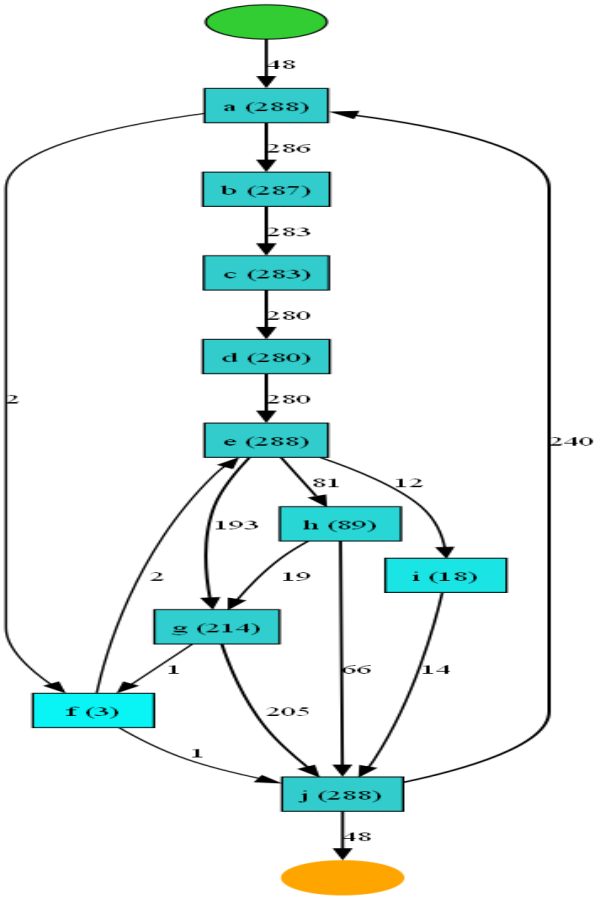


Figure 4-5 High performer's Process model obtained using Heuristic Miner

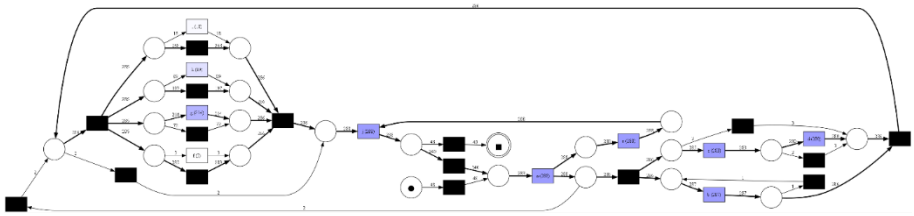


Figure 4-6 High performers Petrinet diagram obtained using Heuriatic Miner

The process models are constructed by utilizing the collective paths navigating by each group of students within LMS. Diagram of each Control-Flow exhibits potential actions and their connections as depicted in a corresponding student log. Visually, the Heuristic Miner generates two artificially created nodes: one named the "source" (positioned at the top in green) signifies the distribution of students' initial activities,

whiles the other, termed the "sink" (located at the bottom in pink), and represents the distribution of students' concluding activities. For example Figure 4-5 represents the source at the top indicating 48 students logged in to the LMS and performed activities in a sequence of view lectures with frequency of (286), watch video lectures (283), download handouts (280) and attempt quiz (280). Here the values present in the edges denote the number of students who initiated a particular activity. In a group of high performer 98% students watch video lectures before attempting quiz. We considered that this group of students performs activities in a guided manner. They followed sequence predefined by instructor i.e., watch video lectures, download handouts and attempt quiz in a row.

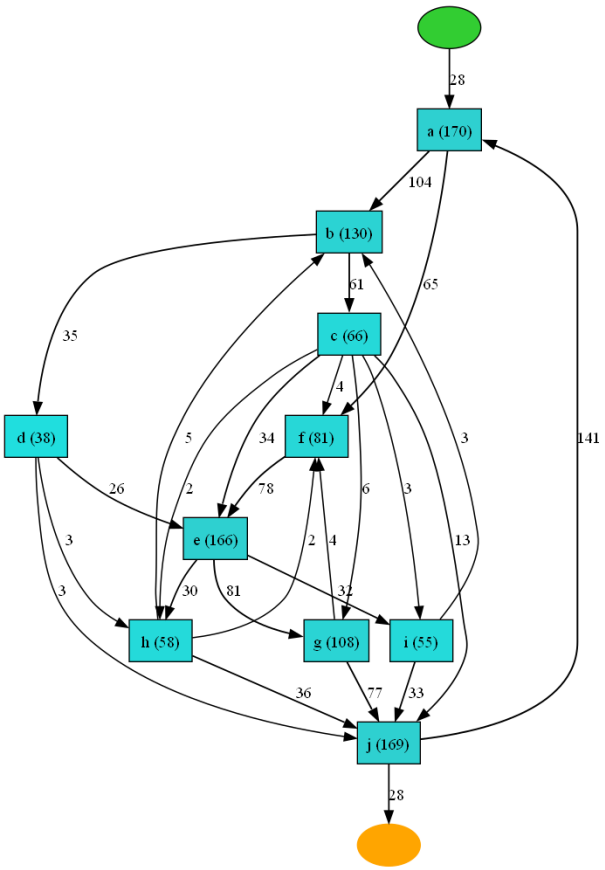


Figure 4-7 Low performer's Process model obtained using Heuristic Miner

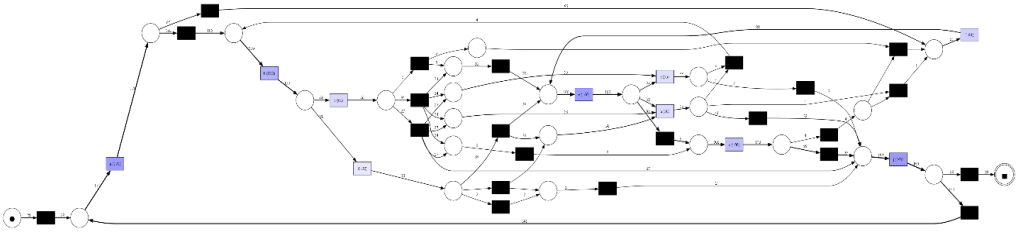


Figure 4-8 High performer's Petrinet diagram obtained using Heuristic Miner

In Figure 4-7 we observed diverse student behavior in low performer group of students. They randomly performed activities in LMS. Here source node at the top indicated that 28 students logged in to the LMS a total of 170 times, starting with viewing lectures with the frequency of (104), and then we observe distribution of paths at node c i.e., watch video lectures with frequency of 61 directed towards 4 students view their progress and 35 students download handouts respectively. Here it is noted that only 35% students watch video lectures before attempting quiz. So as compared to high performer group where we seen pre defined sequence, this cluster of low performers have randomly interact with LMS and perform activities in free manner. Fig Figure 4-6 and Figure 4-8 shows the observed behavior of process model and transitions in detail.

4.3 Inductive Miner:

The inductive miner algorithm process log data in order to generate a comprehensive output by identifying events, their sequences in which they occur and their relationship in various steps including initialization, tracing, segmentation, recursion, and integration of events.

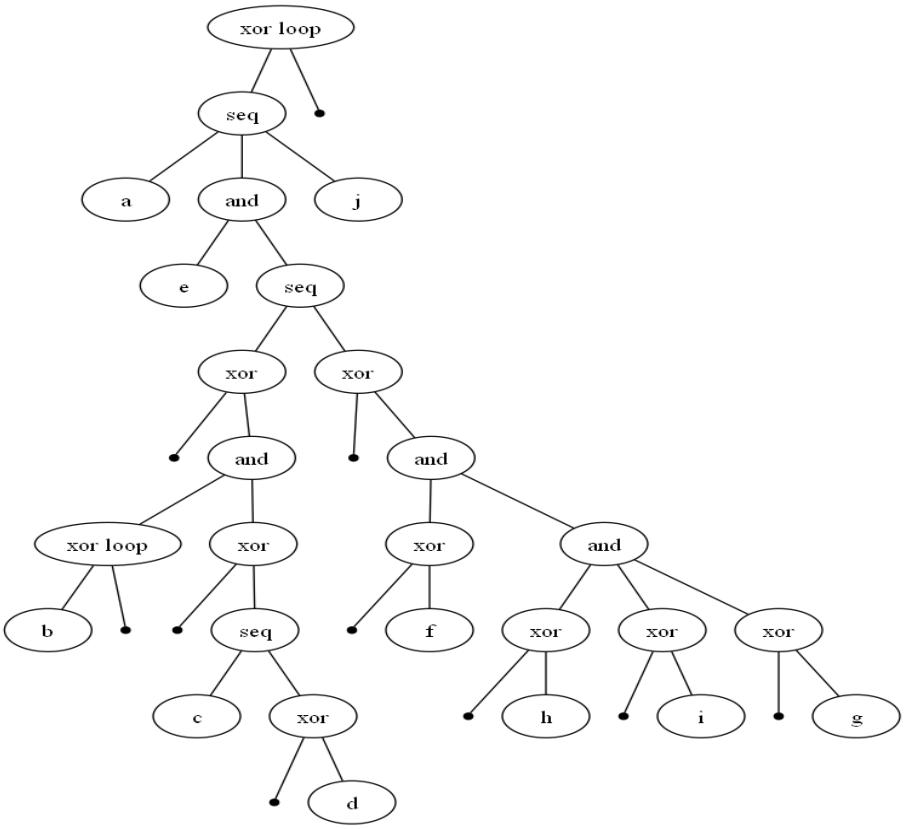


Figure 4-9 High performer's Process model obtained using Inductive Miner

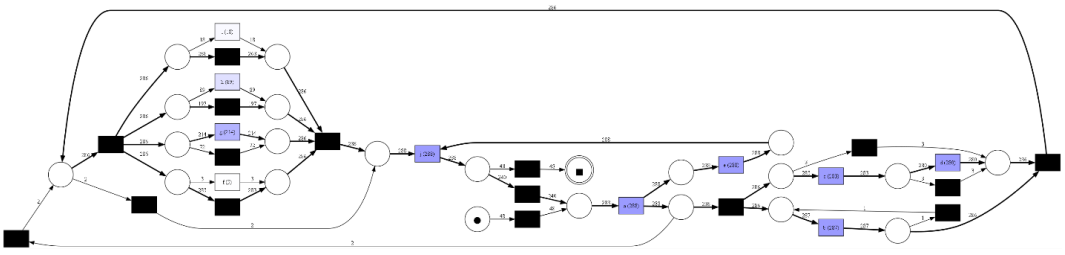


Figure 4-10 High performer's Petrinet obtained using Inductive Miner

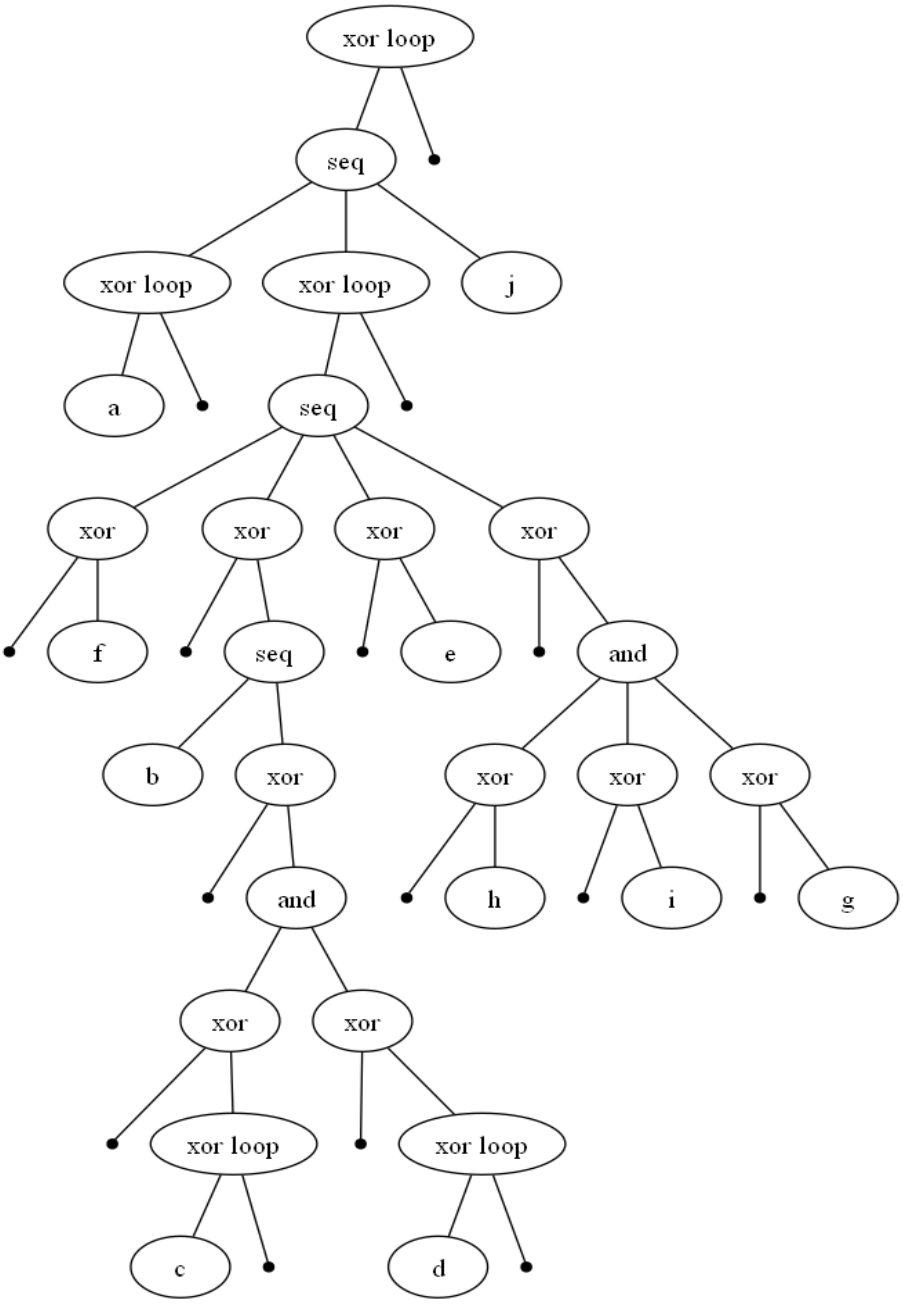


Figure 4-11 Low performer's Process model obtained using Inductive Miner

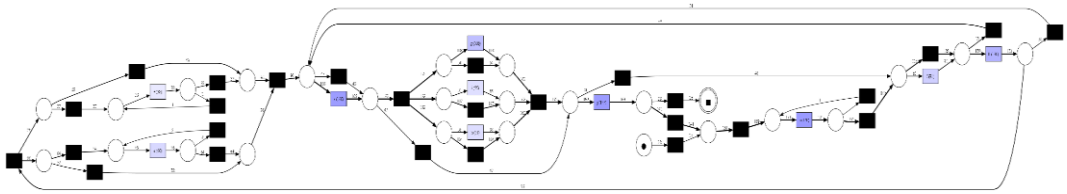


Figure 4-12 Low performer's Petrinet obtained using Inductive Miner

In Figure 4-9 process tree exhibits the whole process from login to logout of student interaction with LMS. Top most node represent the initial action taken by student and then leaf nodes represents the other activities performed by them on LMS. The connection between the nodes illustrates sequences of activities. Here we observed that nodes are arranged on the basis of frequency of activities. Figure 4-11 depicts the diverse sequences than other process tree represented through Figure 4-9**.** Petrinet visualizations in Figure 4-10 and Figure 4-12 visualize the complex process of process tree and demonstrate the observed paths and deviations.

4.4 ILP Miner

The ILP Miner uses Integer Linear Programming to find a model that fits the event log data. It represents the process model as a Petri net, which captures the flow of activities and their relationship in more structured and comprehensive manner. In Figure 4-13 represents student logs of high performer cluster containing 48 students. Here we observed the restricted flow of activities between nodes depicting the students interact with LMS in a guided manner. In contrast Figure 4-14 of low performer cluster containing 28 students, illustrates more diverse and complex behavior of students which deviated from the original path.

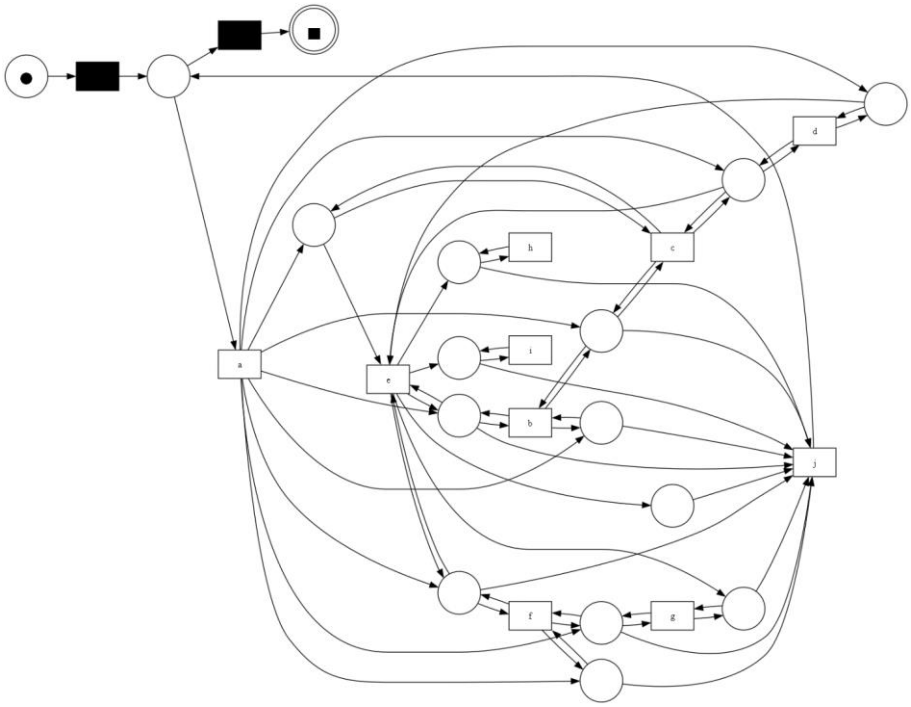


Figure 4-13 High performer's Petrinet obtained using ILP Miner

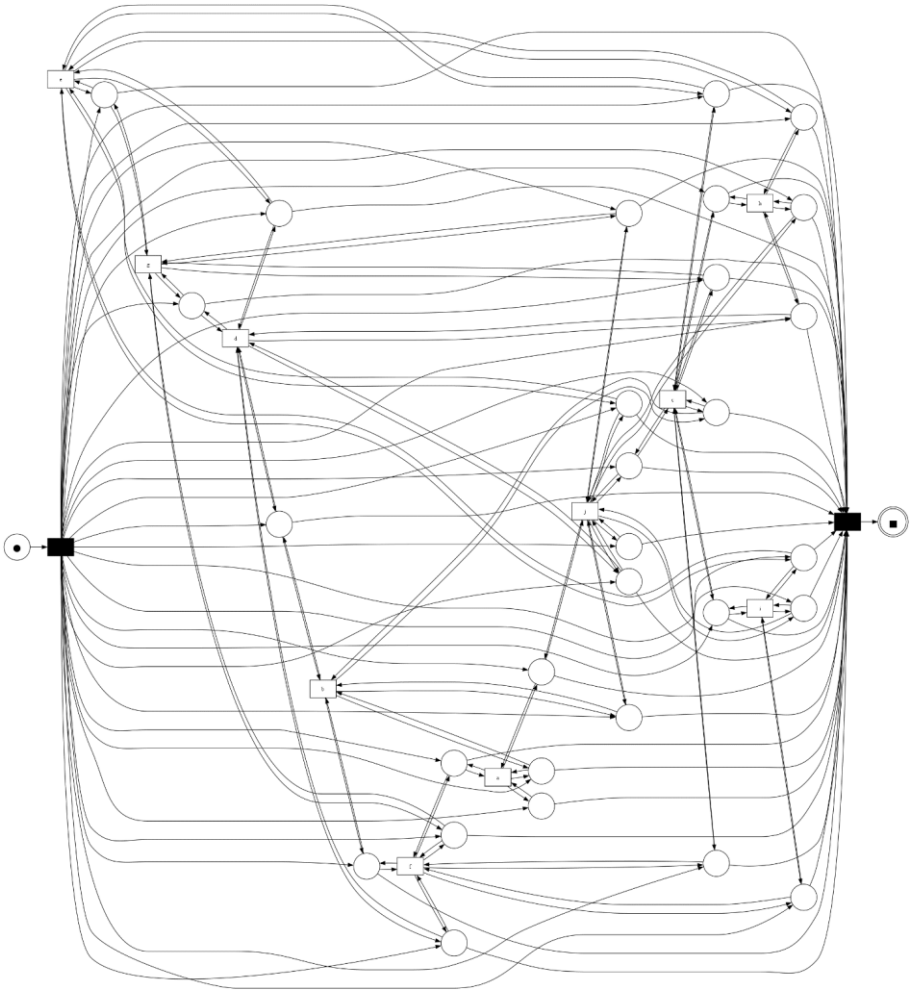


Figure 4-14 Low performer's Petrinet obtained using ILP Miner

4.5 Conformance Checking

Figure 4-15 and Figure 4-16 present the outcomes obtained from the examination of conformance checking of process models generated for high performer's cluster of students, using four distinct algorithms. The IM and ILP miners exhibit a flawless fitness score, indicative of their ability to generate a process model that perfectly corresponds to the genuine event log-based process. Nevertheless, these algorithms are surpassed by heuristic and alpha miner algorithms across all other evaluative metrics of quality.

Heuristic miner yields slightly lower yet competitive fitness scores. In return, alpha miner algorithm yields a process model characterized by the utmost precision score. Furthermore, ILP achieves the highest score in terms of generalization, denoting the model's capacity to maintain precision in capturing similar behaviors while effectively excluding entirely unrelated behaviors. Although, Heuristic attains the highest score for simplicity as well, ensuring a comprehensible and reader-friendly process model. Comparatively, Inductive and ILP miner algorithms gain the highest accuracy score relative to its counterparts. Consequently, based on model accuracy, we opt for the process model generated by Inductive miner and ILP miner to faithfully depict the authentic process for subsequent performance analysis.

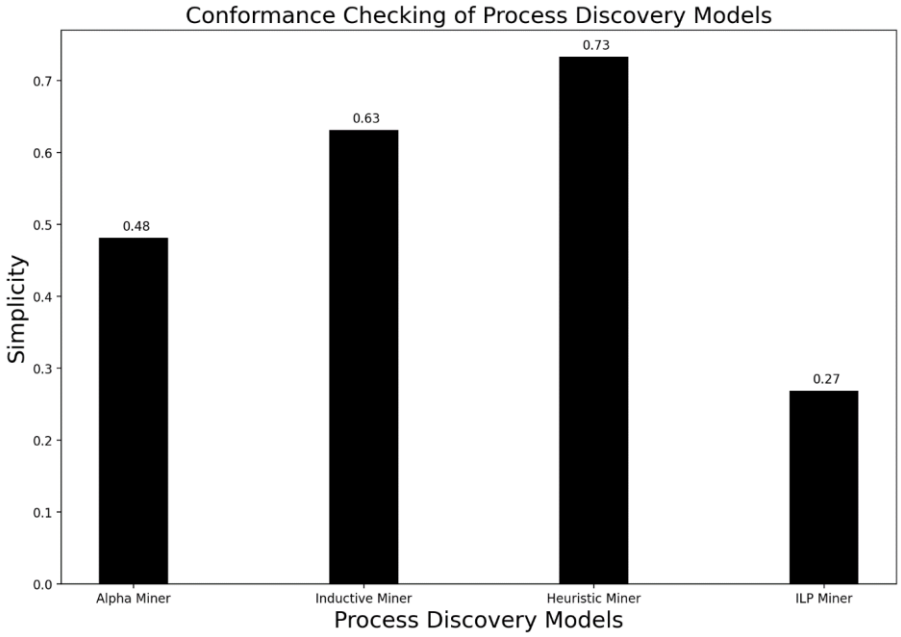


Figure 4-15 Conformance checking of high performers using simplicity

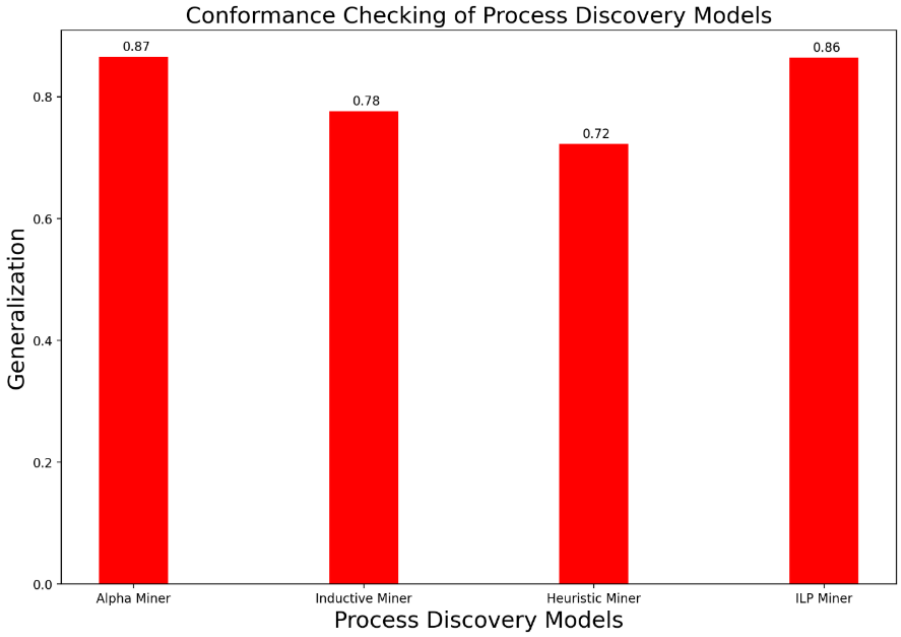


Figure 4-16 Conformance checking of high performers using Generalization

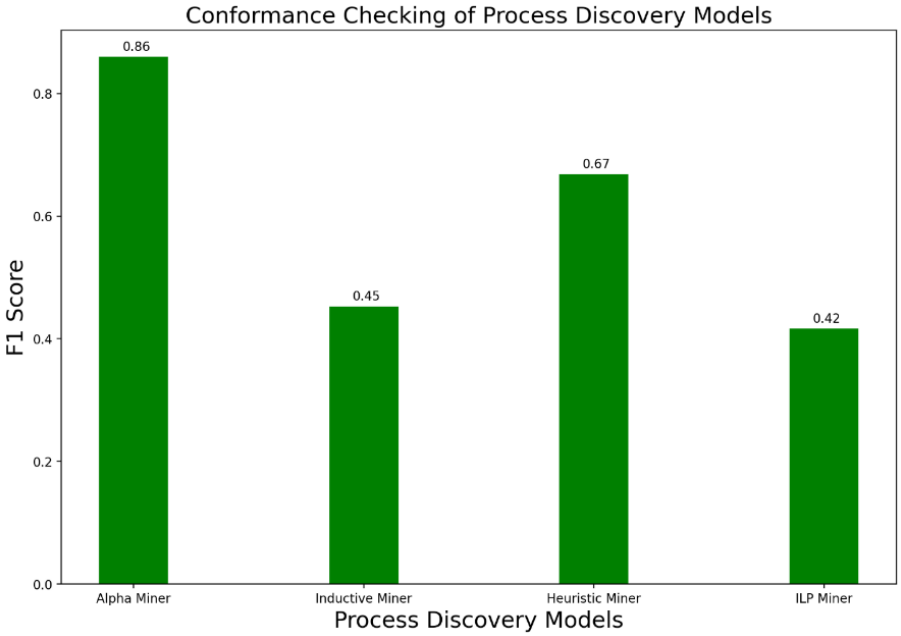


Figure 4-17 Conformance checking of high performers using F1 score

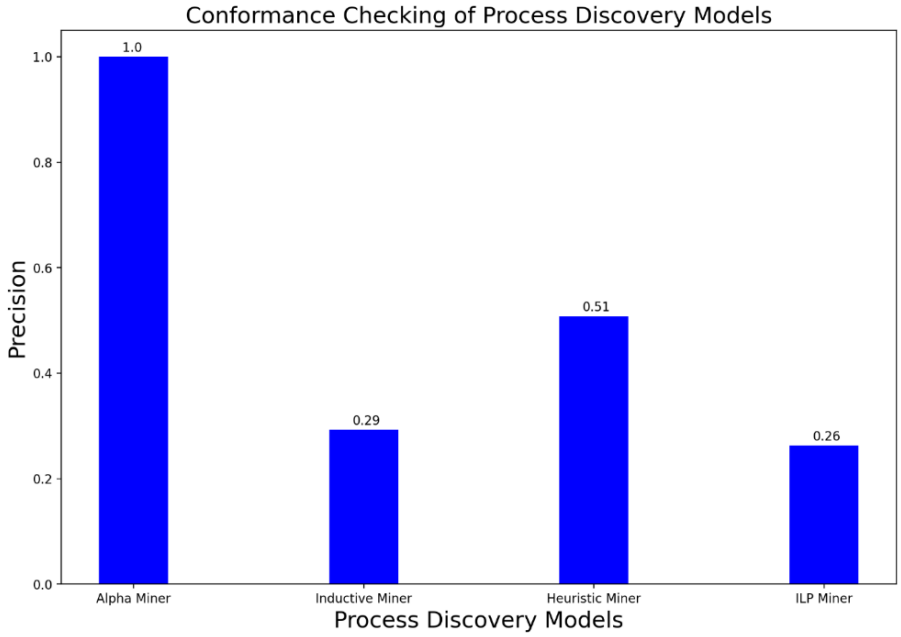


Figure 4-18 Conformance checking of high performers using Precision

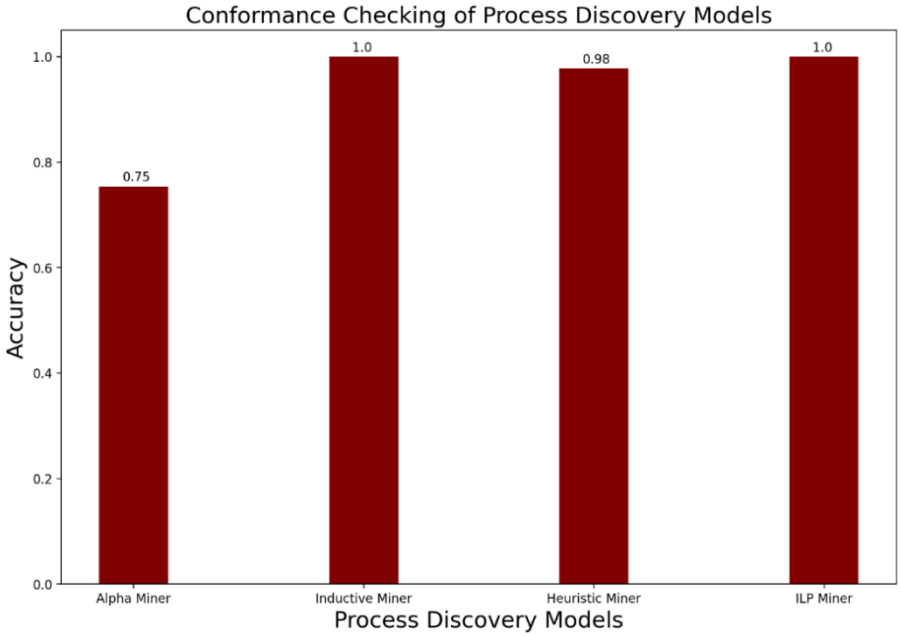


Figure 4-19 Conformance checking of high performers using Accuracy

Conformance Checking
Comparison of Process Discovery Models

	Accuracy	Precision	F1 Score	Generalization	Simplicity
Alpha Miner	0.7538700471997525	1.0	0.8596646580553552	0.8657579000510527	0.48148148148148145
Inductive Miner	1.0	0.2923841442274355	0.45247250290619684	0.776511090466898	0.631578947368421
Heuristic Miner	0.9775564516396833	0.5077234742972905	0.668329719241479	0.7227812838418737	0.7333333333333333
ILP Miner	1.0	0.26282516322935023	0.4162494870742717	0.8640753221596429	0.2692307692307692

Figure 4-20 Summary of conformance checking of high performers

In Figure 4-15 alpha miner shows the highest simplicity value of model obtained from process discovery of low performer group. Figure 4-16 illustrates the alpha and ILP miner as the more generalized as compared to other including inductive miner and heuristic miner. But in Figure 4-17 and Figure 4-18 heuristic miner remains more precise and has highest value of F1 measure. Comparative analyses of all results are shown in Figure 4-20, In contrast to other quality metrics, which we used for the evaluation of process models. We are interested in models which have highest accuracy or fitness so we can seen in Figure 4-19 that ILP miner and inductive miner algorithm shows perfect fitness value which shows that observed behavior is perfectly aligned with original behavior of the event log and provide accurate results.

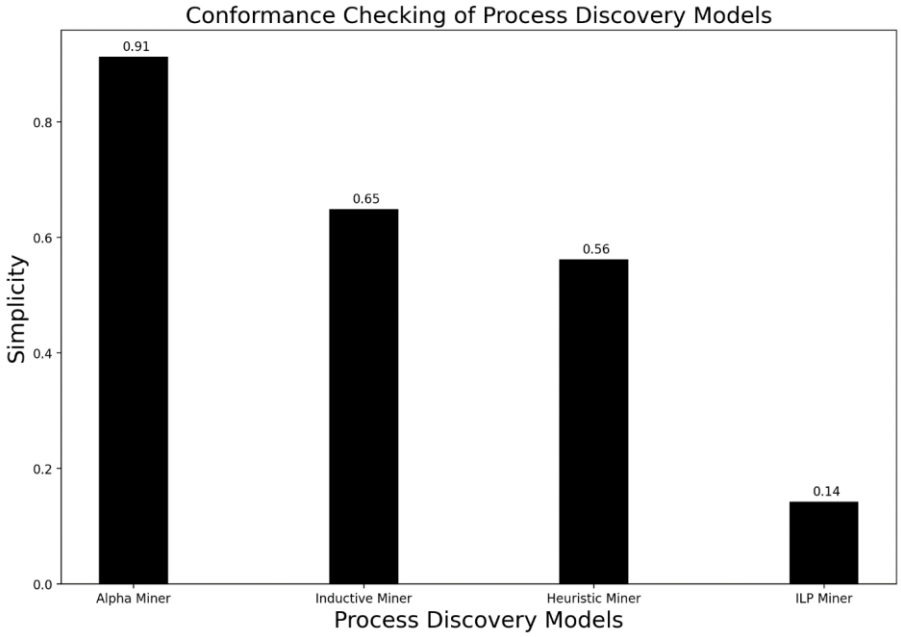


Figure 4-21 Conformance checking of low performers using simplicity

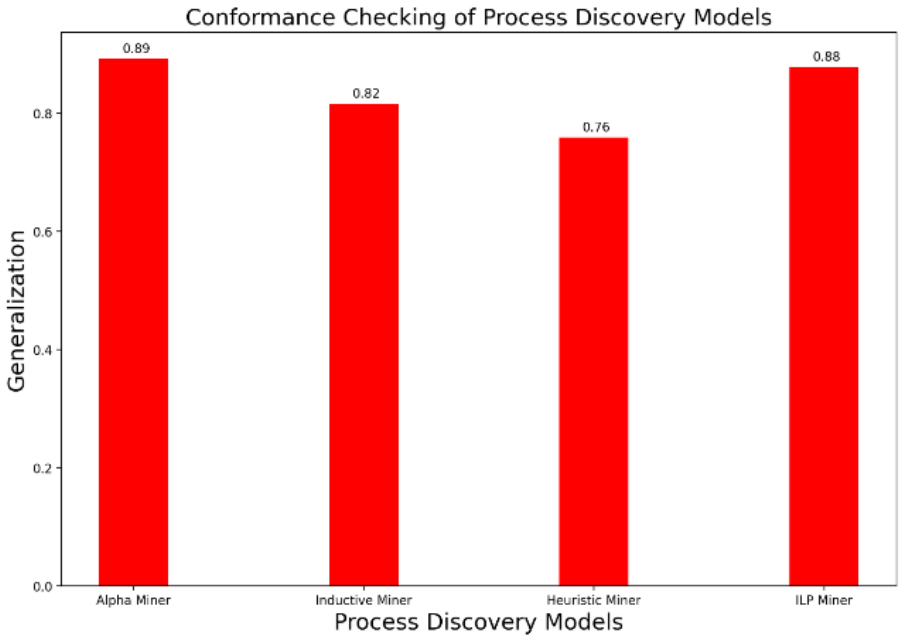


Figure 4-22 Conformance checking of low performers using Generalization

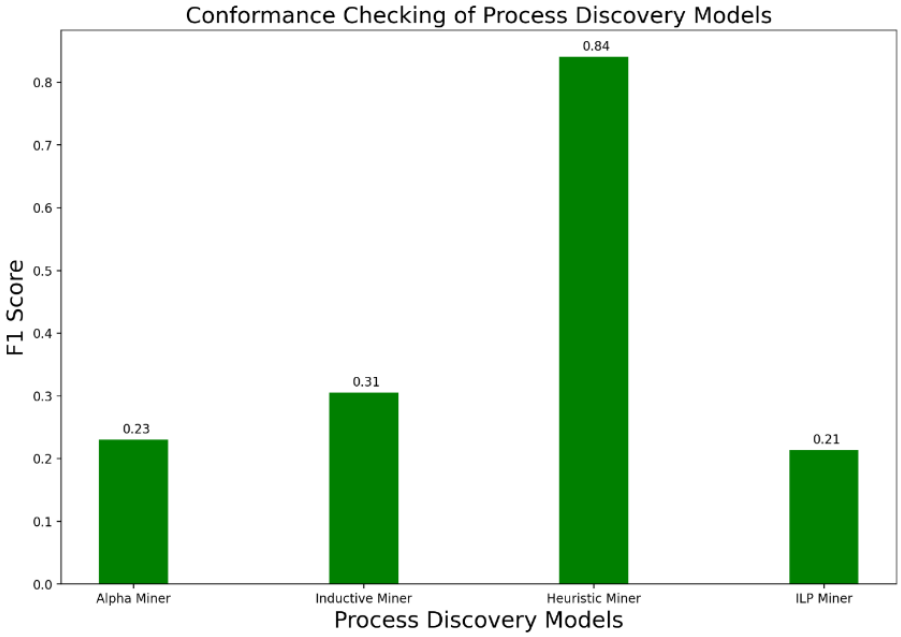


Figure 4-23 Conformance checking of low performers using F1 score

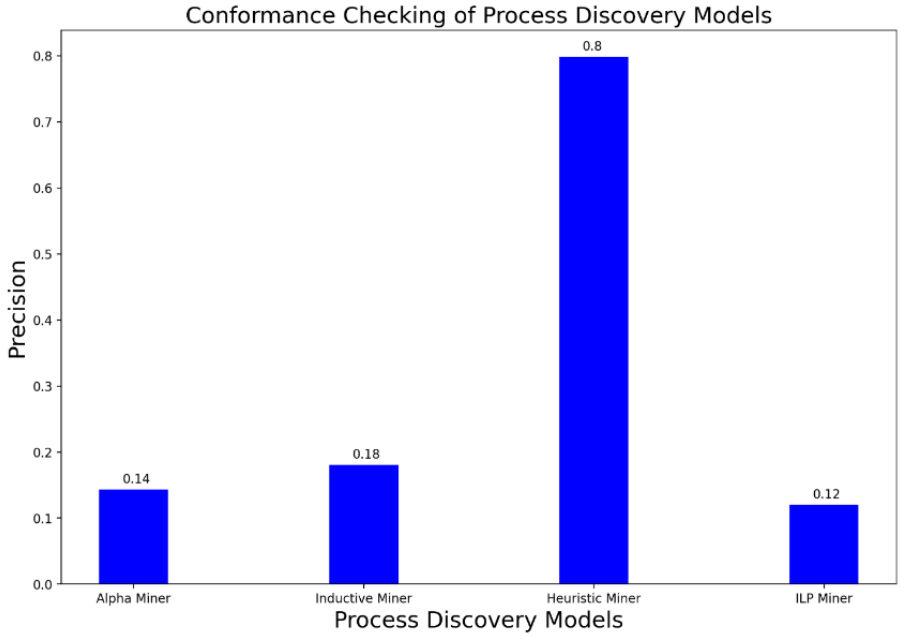


Figure 4-24 Conformance checking of low performers using Precision

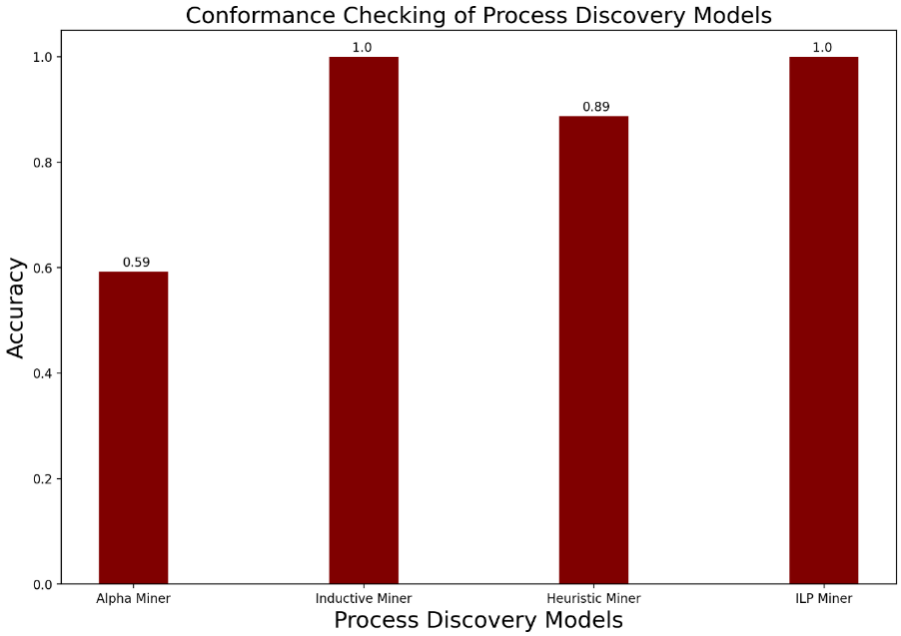


Figure 4-25 Conformance checking of low performers using Accuracy

Conformance Checking
Comparison of Process Discovery Models

	Accuracy	Precision	F1 Score	Generalization	Simplicity
Alpha Miner	0.5926247621660632	0.1428571428571429	0.23021825481013403	0.8922264096252335	0.9130434782608695
Inductive Miner	1.0	0.18043698451743595	0.3057121843589115	0.8160113860009732	0.6494845360824743
Heuristic Miner	0.8874789714011055	0.7983870967741935	0.8405789437259139	0.7594999770266275	0.5625
ILP Miner	1.0	0.11978866474543703	0.2139487003517199	0.878691635270259	0.14285714285714285

Figure 4-26 Summary of Conformance checking of low performers

Derived from the findings, it is observed that the inductive and ILP miner algorithms yield the highest accuracy outcomes for assessing fitness, registering a notable score of 1 as seen in Figure 4-25. In contrast, the alpha miner and heuristic miner algorithm's performance in this regard is comparatively low, producing an accuracy score of 0.59 and 0.89 for low performers cluster and 0.75 and 0.98 in high performers cluster analysis respectively. However, when considering the metrics of precision in Figure 4-24, simplicity in Figure 4-21, F1 measure in Figure 4-23 and generalization in Figure 4-22, the inductive miner algorithm excels as compared to ILP miner algorithm, attaining a precision value of 0.29, simplicity of 0.63, F1 measure score of 0.67 in high performers cluster analysis. While for low performers inductive miner has attain precision value of 0.18, score of simplicity is 0.65, and F1 measure value is 0.31 respectively. Although ILP miner algorithm is more generalized as compared to inductive miner but considering overall quality metrics results in Figure 4-26 we found inductive miner algorithm best in interpreting learning styles of students while interacting with and navigating through LMS.

4.6 GSP Results:

The GSP algorithm was applied to the specified parameters: minimum support of 0.5, window size of 1, minimum gap of 1, and maximum gap of 50. The outcomes reveal that sequential associations featuring substantial support values (proximate to unity) have emerged among pairs of actsivities. This implies that sequences comprising over three activities are discernible even at lower support thresholds. Table IV offers a condensed overview of sequential activities surpassing the established threshold of support. The process of sequential pattern mining offers insights into the procedural pathways adhered to by students in relation to their activities. Here Sequential patterns were obtained from GSP algorithm using SPMF open source java library as shown in Figure 4-27.

Figure 4-27 Output of GSP algorithm

4.6.1 Student Count in each Cluster

Figure 4-28 shows the summary of student logs extracted from LMS which illustrated the clustering of students using K-means cluster based on their interactions with LMS. We have total 76 students registered on LMS, among those 48 students follows pre-defined sequence which is (watch video lecture -> download handouts -> attempt quiz) and other 28 students follows random sequences.

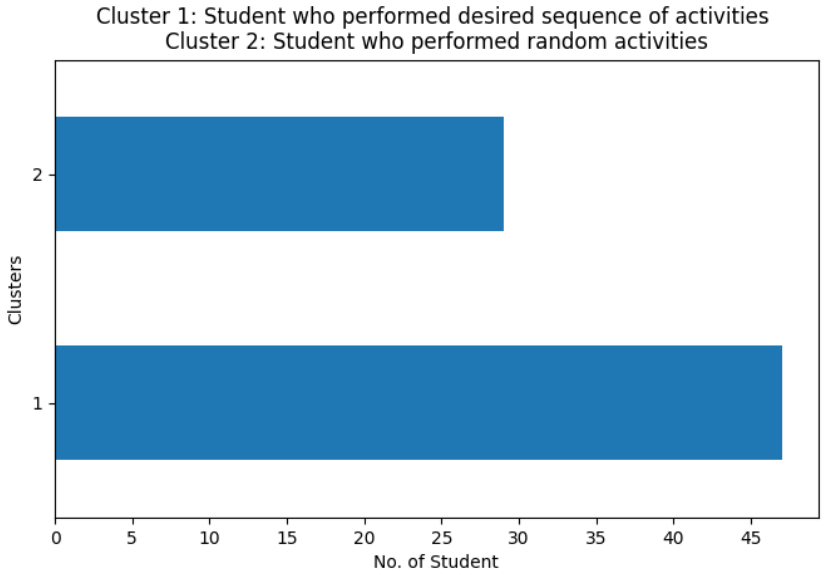


Figure 4-28 student count

4.6.2 Clusters of Students on basis of Performance

We cluster students on the basis of similarity in their interaction patterns along with their grades. Following Figure 4-29 illustrated the overall behavior of students and we can observed that students who followed predefined sequence get better grades as compared to those students who interact randomly with LMS.

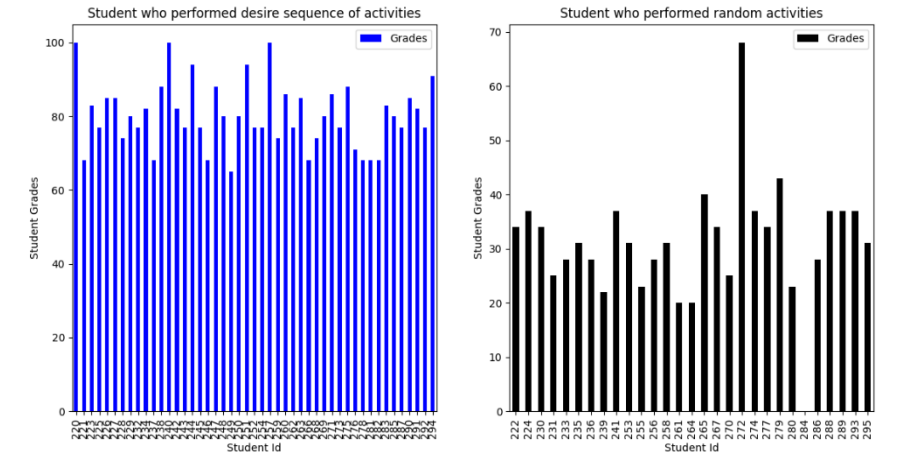


Figure 4-29 Studemt performance based on interaction patterns

4.6.3 Visualization of Student Performance who follows Pre-defined Sequences

In Figure 4-30, normal distribution illustrated the grades of students who interact with LMS in guided manner. Here we can easily seen that their grades lies between 60 to 100 which considers as relatively high as compared to other group shown in Figure 4-31.

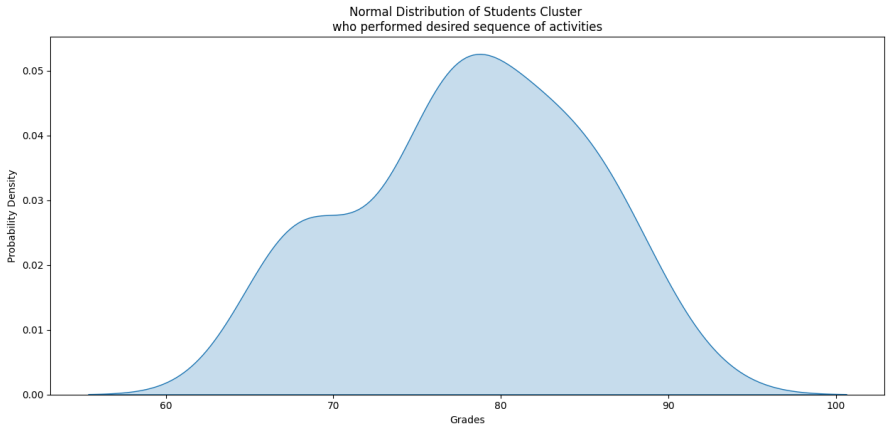


Figure 4-30 Normal distribution of high performer cluster

4.6.4 Visualization of student performance who follows random sequences

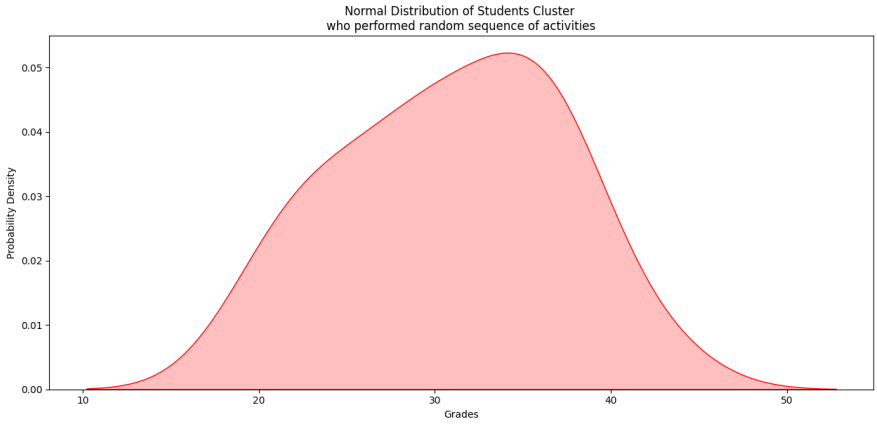


Figure 4-31 Normal distribution of low performer cluster

The above Figure 4-31 depicted the behavior of students who accessed content on LMS randomly having diverse behavior. They deviated from path predefined by instructor and interact with LMS in free manner rather than guided, we observed the clear difference between their performances, such students lies in low performers group.

4.6.5 Student Performance comparison from both clusters

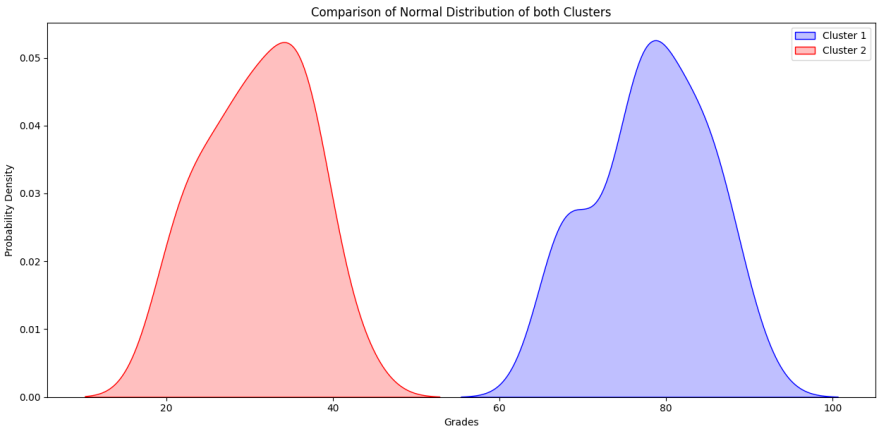


Figure 4-32 Comparison of performance of both clusters

Process mining and sequential pattern mining is powerful techniques for analyzing student performance based on interaction patterns in e-learning environments. These methodologies provide valuable insights into students' learning paths, help identify factors influencing their performance, and inform instructional strategies Figure 4-32 represent the comparative analysis of both groups and illustrate the impact of predefined learning pattern and randomly accessd learning trajectories on student performance. From above results we analyze that students who interact with LMS have diverse behavior but who followed prescribed sequence have get better grades as compared to other group who freely accessed LMS in random fashion. So we can say that predefined learning strategy have good impact on student performance.

CHAPTER 5

CONCLUSION

This chapter discusses the conclusion of our research that focuses on interaction pattern mining in e-learning using process mining and sequential pattern mining in order to measure student performance. We also describe limitation and future work of our study in this section.

We evaluated how interaction pattern mining within students' learning trajectories and addresses the impact of student interaction patterns on their performance in e-learning courses. We noted in the beginning that instructors structure the course sequence based on their didactic and pedagogical strategies, with the intention of guiding students through their learning journey. We also highlighted that in the absence of strict constraints, students might opt for learning paths that diverge from the predefined sequence.

We developed an LMS with which students can interact in both a directed and free manner. We utilized an event log containing 37,405 events, gathered from 76 undergraduate students. Prior to analysis, this log underwent a preprocessing phase. For experiment, we segmented log data into three distinct datasets. To derive statistical insights, we employed the PROM framework. Our investigation entailed the application of four distinct process discovery algorithms namely, Alpha Miner, Heuristic Miner, ILP Miner, and Inductive Miner also GSP algorithm were implemented through scripts based on the PM4PY library. The outcomes of our study revealed that students exhibited unique behaviors while accessing the LMS and engaging in activities. Interestingly, we observed that students who followed a predetermined sequence or interacted with the LMS in a guided manner achieved higher grades compared to their counterparts who navigated the

LMS in a more random fashion. As an outlook, we plan to perform our investigation on a wider range of audience for improved results.

REFERENCES

[1] G. Chung, “Toward the relational management of educational measurement data,” *Teach. Coll. Rec.*, vol. 116, no. 11, 2014, doi: 10.1177/016146811411601115.

[2] B. K. Daniel, “Big data and learning analytics in higher education: Current theory and practice,” *Big Data Learn. Anal. High. Educ. Curr. Theory Pract.*, pp. 1–272, 2016, doi: 10.1007/978-3-319-06520-5.

[3] P. C. Abrami, R. M. Bernard, E. M. Bures, E. Borokhovski, and R. M. Tamim, “Interaction in distance education and online learning: Using evidence and theory to improve practice,” *J. Comput. High. Educ.*, vol. 23, no. 2–3, pp. 82–103, 2011, doi: 10.1007/s12528-011-9043-x.

[4] S. J. J. Leemans, *Process Mining*, vol. 440. 2022.

[5] W. Van Der Aalst *et al.*, “Process mining manifesto,” *Lect. Notes Bus. Inf. Process.*, vol. 99 LNBIP, no. PART 1, pp. 169–194, 2012, doi: 10.1007/978-3-642-28108-2_19.

[6] W. Van der Aalst, *Process mining: Data science in action*. 2016.

[7] U. Cases and L. Reinkemeyer, *Process Mining in Action*. 2020.

[8] K. E. Clayton, F. C. Blumberg, and J. A. Anthony, “Linkages between course status, perceived course value, and students’ preference for traditional versus non-traditional learning environments,” *Comput. Educ.*, vol. 125, no. September 2017, pp. 175–181, 2018, doi: 10.1016/j.compedu.2018.06.002.

[9] E. D. Wagner, “In Support of a Functional Definition of Interaction,” *Am. J. Distance Educ.*, vol. 8, no. 2, pp. 6–29, 1994, doi: 10.1080/08923649409526852.

[10] M. G. Moore, “Editorial: Three Types of Interaction,” *Am. J. Distance Educ.*, vol. 3, no. 2, pp. 1–7, 1989, doi: 10.1080/08923648909526659.

[11] G. Falloon, “Making the connection: Moore’s theory of transactional distance and its relevance to the use of a virtual classroom in postgraduate online teacher

education," *J. Res. Technol. Educ.*, vol. 43, no. 3, pp. 187–209, 2011, doi: 10.1080/15391523.2011.10782569.

[12] R. H. Woods and J. D. Baker, "Interaction and immediacy in online learning," *Int. Rev. Res. Open Distance Learn.*, vol. 5, no. 2, 2004, doi: 10.19173/irrodl.v5i2.186.

[13] T. Pham, V. Thalathoti, and E. Dakich, "Frequency and pattern of learner-instructor interaction in an online English language learning environment in Vietnam," *Australas. J. Educ. Technol.*, vol. 30, no. 6, pp. 686–698, 2014, doi: 10.14742/ajet.608.

[14] D. C. A. Hillman, D. J. Willis, and C. N. Gunawardena, "Learner-Interface Interaction in Distance Education: An Extension of Contemporary Models and Strategies for Practitioners," *Am. J. Distance Educ.*, vol. 8, no. 2, pp. 30–42, 1994, doi: 10.1080/08923649409526853.

[15] E. Lehtinen, "Developing models for distributed problem-based learning: Theoretical and methodological reflection," *Distance Educ.*, vol. 23, no. 1, pp. 109–117, 2002, doi: 10.1080/01587910220124017.

[16] C. E. Wanstreet, "Interaction in Online Learning Environments: A Review of the Literature," *Q. Rev. Distance Educ.*, vol. 7, no. 4, pp. 399–411, 2006.

[17] T. Anderson, "Getting the mix right again: An updated and theoretical rationale for interaction," *Int. Rev. Res. Open Distance Learn.*, vol. 4, no. 2, pp. 126–141, 2003, doi: 10.19173/irrodl.v4i2.149.

[18] R. M. Bernard *et al.*, "A meta-analysis of three types of interaction treatments in distance education," *Rev. Educ. Res.*, vol. 79, no. 3, pp. 1243–1289, 2009, doi: 10.3102/0034654309333844.

[19] D. L. Taylor, "Interactions in Online Courses and Student Academic Success," 2014.

[20] D. R. Garrison, "Self-directed learning: Toward a comprehensive model," *Adult Educ. Q.*, vol. 48, no. 1, pp. 18–33, 1997, doi: 10.1177/074171369704800103.

[21] V. A. Thurmond, "Examination of Interaction Variables as Predictors of Students' Satisfaction And Willingness To Enroll in Future Web-Based Courses While

Controlling for Student Characteristics," *Soc. Inf. Technol. Teach. Educ. Int. Conf.*, pp. 528–531, 2003.

[22] D. Dakic, D. Stefanovic, I. Cosic, T. Lolic, and M. Medojevic, "Business process mining application: A literature review," *Ann. DAAAM Proc. Int. DAAAM Symp.*, vol. 29, no. 1, pp. 866–875, 2018, doi: 10.2507/29th.daaam.proceedings.125.

[23] M. A. Ghazal, O. Ibrahim, and M. A. Salama, "Educational process mining: A systematic literature review," *Proc. - 2017 Eur. Conf. Electr. Eng. Comput. Sci. EECS 2017*, pp. 198–203, 2018, doi: 10.1109/EECS.2017.45.

[24] W. M. P. van der Aalst, *Process Mining*. 2011.

[25] S. Dunzer, M. Stierle, M. Matzner, and S. Baier, "Conformance checking: A state-of-the-art literature review," *ACM Int. Conf. Proceeding Ser.*, 2019, doi: 10.1145/3329007.3329014.

[26] E. M. H. Real, E. Pinheiro Pimentel, L. V. De Oliveira, J. Cristina Braga, and I. Stiubiener, "Educational Process Mining for Verifying Student Learning Paths in an Introductory Programming Course," *Proc. - Front. Educ. Conf. FIE*, vol. 2020-Octob, 2020, doi: 10.1109/FIE44824.2020.9274125.

[27] J. C. Vidal, B. Vázquez-Barreiros, M. Lama, and M. Mucientes, "Recompiling learning processes from event logs," *Knowledge-Based Syst.*, vol. 100, pp. 160–174, 2016, doi: 10.1016/j.knosys.2016.03.003.

[28] J. Munoz-Gama and J. Carmona, "Enhancing precision in Process Conformance: Stability, confidence and severity," *IEEE SSCI 2011 Symp. Ser. Comput. Intell. - CIDM 2011 2011 IEEE Symp. Comput. Intell. Data Min.*, pp. 184–191, 2011, doi: 10.1109/CIDM.2011.5949451.

[29] L. T. Ly, F. M. Maggi, M. Montali, S. Rinderle-Ma, and W. M. P. Van Der Aalst, "A framework for the systematic comparison and evaluation of compliance monitoring approaches," *Proc. - IEEE Int. Enterp. Distrib. Object Comput. Work. EDOC*, pp. 7–16, 2013, doi: 10.1109/EDOC.2013.11.

[30] C. dos S. Garcia *et al.*, "Process mining techniques and applications – A systematic mapping study," *Expert Syst. Appl.*, vol. 133, pp. 260–295, 2019, doi:

10.1016/j.eswa.2019.05.003.

[31] A. Bogarín, R. Cerezo, and C. Romero, "A survey on educational process mining," *Wiley Interdiscip. Rev. Data Min. Knowl. Discov.*, vol. 8, no. 1, 2018, doi: 10.1002/widm.1230.

[32] A. Bolt, M. de Leoni, and W. M. P. van der Aalst, "Scientific workflows for process mining: building blocks, scenarios, and implementation," *Int. J. Softw. Tools Technol. Transf.*, vol. 18, no. 6, pp. 607–628, 2016, doi: 10.1007/s10009-015-0399-5.

[33] H. Alqaheri and M. Panda, "An Education Process Mining Framework: Unveiling Meaningful Information for Understanding Students' Learning Behavior and Improving Teaching Quality," *Inf.*, vol. 13, no. 1, 2022, doi: 10.3390/info13010029.

[34] A. Bogarín, R. Cerezo, and C. Romero, "Discovering learning processes using inductive miner: A case study with learning management systems (LMSs)," *Psicothema*, vol. 30, no. 3, pp. 322–329, 2018, doi: 10.7334/psicothema2018.116.

[35] M. C. Rodríguez, M. L. Nistal, and F. A. M. Fonte, "Exploring the Application of Process Mining to Support Self-Regulated Learning," *2018 IEEE Glob. ...*, pp. 1772–1780, 2018.

[36] A. Bogarín, C. Romero, R. Cerezo, and M. Sánchez-Santillán, "Clustering for improving Educational process mining," *ACM Int. Conf. Proceeding Ser.*, pp. 11–15, 2014, doi: 10.1145/2567574.2567604.

[37] C. Romero, S. Ventura, and E. García, "Data mining in course management systems: Moodle case study and tutorial," *Comput. Educ.*, vol. 51, no. 1, pp. 368–384, 2008, doi: 10.1016/j.compedu.2007.05.016.

[38] W. van der Aalst *et al.*, "Preface: 9th International workshop on business process intelligence (BPI 2013)," *Lect. Notes Bus. Inf. Process.*, vol. 171 171 LN, no. May, pp. 0–12, 2014, doi: 10.1007/978-3-319-06257-0.

[39] E. Rojas, J. Munoz-Gama, M. Sepúlveda, and D. Capurro, "Process mining in healthcare: A literature review," *J. Biomed. Inform.*, vol. 61, pp. 224–236, 2016,

doi: 10.1016/j.jbi.2016.04.007.

[40] B. F. Van Dongen, A. K. A. De Medeiros, H. M. W. Verbeek, A. J. M. M. Weijters, and W. M. P. Van Der Aalst, "The ProM framework: A new era in process mining tool support," *Lect. Notes Comput. Sci.*, vol. 3536, no. i, pp. 444–454, 2005, doi: 10.1007/11494744_25.

[41] A. K. A. De Medeiros *et al.*, "Process mining based on clustering: A quest for precision," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 4928 LNCS, pp. 17–29, 2008, doi: 10.1007/978-3-540-78238-4_4.

[42] A. Corallo, M. Lazoi, and F. Striani, "Process mining and industrial applications: A systematic literature review," *Knowl. Process Manag.*, vol. 27, no. 3, pp. 225–233, 2020, doi: 10.1002/kpm.1630.

[43] L. K. M. Poon, S. C. Kong, M. Y. W. Wong, and T. S. H. Yau, "Mining sequential patterns of students' access on learning management system," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10387 LNCS, pp. 191–198, 2017, doi: 10.1007/978-3-319-61845-6_20.

[44] Y. Zhang and L. Paquette, "Sequential Pattern Mining in Educational Data: The Application Context, Potential, Strengths, and Limitations," pp. 219–254, 2023, doi: 10.1007/978-981-99-0026-8_6.

[45] T. Slimani and A. Lazzez, "Sequential Mining: Patterns and Algorithms Analysis," 2013.

[46] J. S. Yoo, Y. S. Woo, and S. J. Park, "Mining Course Trajectories of Successful and Failure Students: A Case Study," *Proc. - 2017 IEEE Int. Conf. Big Knowledge, ICBK 2017*, pp. 270–275, 2017, doi: 10.1109/ICBK.2017.55.

[47] E. M. H. Real, E. P. Pimentel, and J. C. Braga, "Analysis of Learning Behavior in a Programming Course using Process Mining and Sequential Pattern Mining," *Proc. - Front. Educ. Conf. FIE*, vol. 2021-Octob, 2021, doi: 10.1109/FIE49875.2021.9637146.

[48] P. Mukala, J. Buijs, and W. Van Der Aalst, "Uncovering Learning Patterns in a

MOOC through Conformance Alignments," *BPM reports*, vol. 1509, pp. 1–20, 2015.

[49] B. A. N. Cenka, H. B. Santoso, and K. Junus, "Analysing student behaviour in a learning management system using a process mining approach," *Knowl. Manag. E-Learning*, vol. 14, no. 1, pp. 62–80, 2022, doi: 10.34105/j.kmel.2022.14.005.

[50] P. Alvarez, J. Fabra, S. Hernandez, and J. Ezpeleta, "Alignment of teacher's plan and students' use of LMS resources. Analysis of Moodle logs," *2016 15th Int. Conf. Inf. Technol. Based High. Educ. Training, ITHET 2016*, 2016, doi: 10.1109/ITHET.2016.7760720.

[51] M. J. Gomez, J. A. Ruipérez-Valiente, P. A. Martinez, and Y. J. Kim, "Exploring the Affordances of Sequence Mining in Educational Games," *ACM Int. Conf. Proceeding Ser.*, pp. 648–654, 2020, doi: 10.1145/3434780.3436562.

[52] Á. Herrero *et al.*, "Preface," *Adv. Intell. Syst. Comput.*, vol. 239, pp. v–vi, 2014, doi: 10.1007/978-3-319-01854-6.

[53] R. Cerezo, A. Bogarín, M. Esteban, and C. Romero, "Process mining for self-regulated learning assessment in e-learning," *J. Comput. High. Educ.*, vol. 32, no. 1, pp. 74–88, 2020, doi: 10.1007/s12528-019-09225-y.

[54] S. Chanifah, R. Andreswari, and R. Fauzi, "Analysis of Student Learning Pattern in Learning Management System (LMS) using Heuristic Mining a Process Mining Approach," *Proceeding - ICERA 2021 2021 3rd Int. Conf. Electron. Represent. Algorithm*, pp. 121–125, 2021, doi: 10.1109/ICERA53111.2021.9538654.

[55] W. Premchaiswadi, P. Porouhan, and N. Premchaiswadi, "Process Modeling, Behavior Analytics and Group Performance Assessment of e-Learning Logs Via Fuzzy Miner Algorithm," *Proc. - Int. Comput. Softw. Appl. Conf.*, vol. 2, pp. 304–309, 2018, doi: 10.1109/COMPSAC.2018.10247.

[56] S. Nakamura, K. Nozaki, Y. Morimoto, and Y. Miyadera, "Sequential pattern mining method for analysis of programming learning history based on the learning process," *2014 Int. Conf. Educ. Technol. Comput. ICETC 2014*, pp. 55–60, 2014, doi: 10.1109/ICETC.2014.6998902.

[57] D. E. Aktas and M. S. Aktas, “Sequential Rule Mining on the Student Behavior Data of an E-Learning Platform in the Field of Financial Sciences: Case Study,” *3rd Int. Conf. Electr. Commun. Comput. Eng. ICECCE 2021*, no. June, pp. 12–13, 2021, doi: 10.1109/ICECCE52056.2021.9514230.

[58] Y. Tomaylla Quispe, O. Gutierrez Aguilar, and A. Gutierrez Aguilar, “Student interaction and satisfaction level with a LMS subject at a higher education institution,” *Proc. - 2020 3rd Int. Conf. Incl. Technol. Educ. CONTIE 2020*, pp. 35–40, 2020, doi: 10.1109/CONTIE51334.2020.00015.

[59] F. Wafda, T. Usagawa, and E. Mahendrawathi, “Systematic Literature Review on Process Mining in Learning Management System,” *Proc. 2022 IEEE Int. Conf. Ind. 4.0, Artif. Intell. Commun. Technol. IAICT 2022*, pp. 160–166, 2022, doi: 10.1109/IAICT55358.2022.9887428.

[60] W. van der Aalst, “Process mining,” *ACM SIGKDD Explor. Newsl.*, vol. 13, no. 2, pp. 45–49, 2012, doi: 10.1145/2207243.2207251.

[61] W. Van Der Aalst, “Process mining: Overview and opportunities,” *ACM Trans. Manag. Inf. Syst.*, vol. 3, no. 2, pp. 1–17, 2012, doi: 10.1145/2229156.2229157.

[62] L. K. M. Poon, S. C. Kong, T. S. H. Yau, M. Wong, and M. H. Ling, “Learning analytics for monitoring students participation online: Visualizing navigational patterns on learning management system,” *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 10309 LNCS, pp. 166–176, 2017, doi: 10.1007/978-3-319-59360-9_15.

[63] S. ElAtia, D. Ipperciel, and O. R. Zaïane, “Educational Process Mining: a Tutorial and Case Study Using Moodle Data Sets on Big Data and Text Mining in the Humanities Finding Predictors in Higher Education Educational Data Mining: a Mooc Experience Data Mining and Action Research At the Intersectio,” 2016.

[64] J. P. Salazar-Fernandez, M. Sepúlveda, and J. Munoz-Gama, “Describing educational trajectories of engineering students in individual high-failure rate courses that lead to late dropout,” *CEUR Workshop Proc.*, vol. 2425, pp. 39–48, 2019.

[65] A. Azeta, F. Agono, F. Adesola, V. Nwaocha, and S. Tjiraso, “A Process Mining Framework for Analysing Students’ Behaviours Using Digital Twin,” *SSRN Electron. J.*, pp. 1–19, 2023, doi: 10.2139/ssrn.4331450.

[66] A. R. Racca, E. Sulis, and S. Capecchi, “Behavioral Web Tracking in e-Learning: An Educational Process Mining Application,” *Proc. Int. Conf. Inf. Vis.*, vol. 2022-July, no. Iv, pp. 269–274, 2022, doi: 10.1109/IV56949.2022.00053.

[67] V. Latypova, “Work with Free Response Implementation Process Analysis Based on Sequential Pattern Mining in Engineering Education,” *2022 6th Int. Conf. Inf. Technol. Eng. Educ. Inforino 2022 - Proc.*, pp. 1–4, 2022, doi: 10.1109/Inforino53888.2022.9782969.

[68] Y. Wang, T. Li, C. Geng, and Y. Wang, “Recognizing patterns of student’s modeling behaviour patterns via process mining,” *Smart Learn. Environ.*, vol. 6, no. 1, 2019, doi: 10.1186/s40561-019-0097-y.

[69] W. Intayoad, C. Kamyod, and P. Temdee, “Process mining application for discovering student learning paths,” *3rd Int. Conf. Digit. Arts, Media Technol. ICDAMT 2018*, pp. 220–224, 2018, doi: 10.1109/ICDAMT.2018.8376527.

[70] A. H. Cairns, B. Gueni, M. Fhima, A. Cairns, and E. Al., “Process Mining in the Education Domain,” *Int. J. Adv. Intell. Syst.*, vol. 08, no. 1 & 2, pp. 219–232, 2015.

[71] S. J. J. Leemans, E. Poppe, and M. T. Wynn, “Directly follows-based process mining: Exploration & a case study,” *Proc. - 2019 Int. Conf. Process Mining, ICPM 2019*, pp. 25–32, 2019, doi: 10.1109/ICPM.2019.00015.

[72] P. Mukala, J. Buijs, M. Leemans, and W. Van Der Aalst, “Learning analytics on coursera event data: A proceb mining approach,” *CEUR Workshop Proc.*, vol. 1527, pp. 18–32, 2015.

[73] J. A. Martínez-Carrascal, E. Valderrama, and T. Sancho-Vinuesa, “Combining clustering and sequential pattern mining to detect behavioral differences in log data: Conceptualization and case study,” *CEUR Workshop Proc.*, vol. 2671, pp. 25–38, 2020.

[74] R. Wang and O. R. Zaïane, “Discovering Process in Curriculum Data to Provide

Recommendation," *Proceeding 8th Int. Conf. Educ. Data Mining, EDM15*, pp. 580–581, 2015.

[75] J. Saint, D. Gašević, and A. Pardo, *Detecting Learning Strategies Through Process Mining*, vol. 11082 LNCS. Springer International Publishing, 2018.

[76] W. C. Shih, "Mining Sequential Patterns to Explore Users' Learning Behavior in a Visual Programming App," *Proc. - Int. Comput. Softw. Appl. Conf.*, vol. 2, pp. 126–129, 2018, doi: 10.1109/COMPSAC.2018.10216.

[77] R. Divya Sri and M. M. Patil, "Study of Learners Behaviour in virtual learning environment using Process Mining," *Proc. CONECCT 2021 7th IEEE Int. Conf. Electron. Comput. Commun. Technol.*, pp. 1–6, 2021, doi: 10.1109/CONECCT52877.2021.9622641.

[78] G. Özdağoğlu, G. Z. Öztaş, and M. Çağliyangil, "An application framework for mining online learning processes through event-logs," *Bus. Process Manag. J.*, vol. 25, no. 5, pp. 860–886, 2019, doi: 10.1108/BPMJ-10-2017-0279.

[79] A. Berti, S. J. Van Zelst, W. M. P. Van Der Aalst, and F. Gesellschaf, "Process mining for python (PM4py): Bridging the gap between process- And data science," *CEUR Workshop Proc.*, vol. 2374, pp. 13–16, 2019.

[80] P. Ceravolo, E. Damiani, M. Torabi, and S. Barbon, "Toward a new generation of log pre-processing methods for process mining," *Lect. Notes Bus. Inf. Process.*, vol. 297, pp. 55–70, 2017, doi: 10.1007/978-3-319-65015-9_4.

[81] W. Van De Aalst, "Process discovery: Capturing the invisible," *IEEE Comput. Intell. Mag.*, vol. 5, no. 1, pp. 28–41, 2010, doi: 10.1109/MCI.2009.935307.

[82] E. Gupta, "Process mining algorithms," *Int. J. Adv. Res. Sci. Eng.*, vol. 3, no. 11, pp. 401–412, 2014.

[83] W. Wien, "Design and Implementation of an Integrated Data Pipeline for Combining Process- and Text-Mining towards Optimizing Human Learning in Business Processes."

[84] M. D. Ivanka, R. Andreswari, and R. Fauzi, "Bottleneck Analysis of Lectures Grades Input Process at Information System Academic using Inductive Miner,"

2021 Int. Semin. Mach. Learn. Optim. Data Sci. ISMODE 2021, pp. 178–183, 2022, doi: 10.1109/ISMODE53584.2022.9742796.

[85] R. Srikant and R. Agrawal, “Mining Quantitative Association Rules in Large Relational Tables,” *SIGMOD Rec. (ACM Spec. Interes. Gr. Manag. Data)*, vol. 25, no. 2, pp. 1–12, 1996, doi: 10.1145/235968.233311.