Permutation-Based Significance Testing for Temporal Sequential Patterns in Learning Event Logs

Tianwei Peng
South China Normal University
2024020894@m.scnu.edu.cn

Jinwang Song
South China Normal University
2427695673@qq.com

Sijing Yu
South China Normal University
643117997@qq.com

Sijie Zhang
South China Normal University
zhangsijie1003@163.com

Yingbin Zhang
South China Normal University
zyingbin@m.scnu.edu.cn

ABSTRACT

Sequential pattern mining is widely used to analyze learning event logs, but support-based mining often produces many candidates whose statistical reliability is unclear. Existing statistical significance pattern mining approaches typically focus on event order and do not incorporate inter-event time intervals. To address this gap, we propose a permutation-testing framework for temporal sequential patterns. The method first mines candidate patterns and then evaluates each pattern's instance value against empirical null distributions generated by two permutation techniques: fixed-time permutation, which shuffles the event order but preserving the time gaps between positions in a sequence, and event-and-interval synchronized permutation, which shuffles events together with their following time gaps. We applied the framework to three learning event sequence datasets. Across datasets, permutation testing removed a substantial subset of frequent patterns, showing that support alone may overestimate meaningful temporal patterns. Longer patterns were generally more likely to remain significant, whereas shorter patterns were more often removed. In addition, removed patterns tended to be composed of more frequent event types. These findings suggest that the proposed permutation-based filtering can improve the interpretability of temporal sequential pattern mining in educational data.

Keywords

Temporal pattern, sequential pattern mining, permutation test, learning process

1. INTRODUCTION

In digital learning environments, students continuously generate rich streams of time-stamped process data through behaviors such as attempt questions, accessing resources, and participating in discussions. These behavioral traces offer valuable opportunities to understand learning processes and identify mechanisms underlying differences in engagement, strategy use, and learning outcomes. Accordingly, methods that can detect recurring structures in behavioral sequences have become increasingly important for understanding authentic learning processes. Among these methods, sequential pattern mining has been widely used to identify recurring sequential combinations of student behaviors [1-2]. Within this line of work, frequent sequential pattern mining is one of the most used approaches because it can efficiently discover patterns that recur across many sequences[3]. However, traditional frequent pattern mining relies primarily on support thresholds, typically determining whether a pattern occurs at least once within a sequence. Although this approach is computationally convenient, it has important limitations. In particular, it does not account for the possibility that some patterns may be frequent simply by contingency or chance rather than reflecting genuinely behavioral regularities.

To address this limitation, recent studies have begun to move beyond support alone by examining the statistical significance or reliability of discovered sequential patterns [4]. For example, Zhang et al. 错误!未找到引用源。 proposed evaluating whether a pattern occurs significantly more often than expected by chance. This line of work has improved the rigor of sequential pattern mining by examining whether it is statistically sound. Nevertheless, existing statistical significance pattern mining approaches focus primarily on ordered event sequences and positional relationships, such as positional gap constraints between events, rather than on actual elapsed time between events. As a result, they remain limited to patterns defined by order information, without incorporating real time information.

This limitation is particularly important in educational settings, where the meaning of a behavioral sequence often depends not only on event order but also on the time intervals between events. The same sequence of events may reflect very different learning behaviors or engagement states when it unfolds over different durations [6]. A student who revisits a resource immediately after answering incorrectly may reflect a different process from one who does so after a longer delay, even if the event order is identical. For this reason, relying solely on event order is insufficient for fully characterizing authentic learning processes [7]. Therefore, time-aware sequence pattern mining has gained increasing attention, with recent studies emphasizing the importance of duration and temporal intervals in pattern representation [8-9].

Tianwei Peng, Jinwang Song, Sijing Yu, Sijie Zhang, and Yingbin Zhang. Permutation-Based Significance Testing for Temporal Sequential Patterns in Learning Event Logs. In Anthony Botelho, Maria Mercedes T. Rodrigo, Adish Singla, Hiroaki Ogata, Hyojeong So, and Young Hoan Cho (eds.) Proceedings of the 19th International Conference on Educational Data Mining, Seoul, Republic of Korea, June, 2026, pp. 732–736. International Educational Data Mining Society (2026).

https://doi.org/10.5281/zenodo.21039724

Nevertheless, two important advances in the literature have largely developed separately. On the one hand, statistically significant sequential pattern mining has improved the reliability of pattern discovery, but it typically does so without incorporating real temporal information. On the other hand, time-aware sequence mining incorporates temporal intervals, but most studies have focused on discovering patterns rather than testing whether those patterns are statistically reliable. This disconnect leaves an important methodological gap. In educational behavioral data, frequently occurring time-aware patterns are not necessarily meaningful. Some may simply reflect incidental arrangements or artifacts of interaction structure. Without statistical testing, it remains difficult to determine which discovered temporal patterns genuinely reflect structured learning behaviors and which are likely to be noise.

To address this gap, the current study proposes applying permutation testing to evaluate the statistical significance of the discovered temporal patterns. Permutation testing provides a random reference distribution against which observed pattern occurrences can be evaluated, making it possible to distinguish patterns that occur more often than expected by chance from those that do not [5]. In this way, the study extends significance-based sequential pattern analysis from order-only patterns to temporal patterns. The goal is to identify temporal patterns that are frequent and statistically significant to provide a more rigorous basis for interpreting learning processes in digital environments. We evaluated the proposed method via three research questions (RQ) in three real datasets:

RQ1: How many frequent temporal patterns mined by sequential pattern mining would be identified as statistically significant by permutation testing?

RQ2: Whether statistically significant and removed (not significant) patterns differ in their length distributions?

RQ3: Whether statistically significant and removed patterns differ in the frequencies of their constituent event types?

2. METHODOLOGY

Figure 1 shows the workflow of the proposed permutation-testing procedure for evaluating the statistical significance of temporal sequential patterns. The procedure is adopted from the permutation testing in [5] by incorporating temporal information. It is independent of any specific temporal pattern mining algorithm. In Step 1, a temporal pattern mining approach is applied to learning event sequences to generate candidate patterns. These candidates are then evaluated in the subsequent permutation test.

In Step 2, permutation testing is used to generate an empirical null distribution for each candidate pattern. For each pattern, its observed instance value is first computed from the original data. The learner sequences are then repeatedly permuted under a specified null model, and the instance value of the same candidate is recomputed after each permutation. This produces a reference distribution showing the pattern's expected instance value in sequences where the learning events are randomly arranged.

We proposed two permutation techniques representing different temporal assumptions. The fixed-time permutation keeps the event times at each position unchanged and permutes only event types. The event-and-interval synchronized permutation permutes each event together with its following inter-event interval, thereby preserving local event–interval pairings.

Finally, we compare the observed instance each candidate pattern is compared with its permutation distribution to obtain permutation-based *p*-values. Because many candidate patterns are tested simultaneously, *p*-values are adjusted using the Benjamini–Yekutieli procedure to control the false discovery rate. Patterns that remain significant after correction are retained as statistically significant temporal patterns.

3. EMPIRICAL DATA ANALYSES

3.1 Datasets

This study analyzed event log datasets from three digital learning environments: Aqualab, LiRuYun, and iSnap[1]. Aqualab was a webpage educational game. LiRuYun was a learning management system. iSnap was a block-based programming platform. Table 1 displays the characteristics of these datasets.

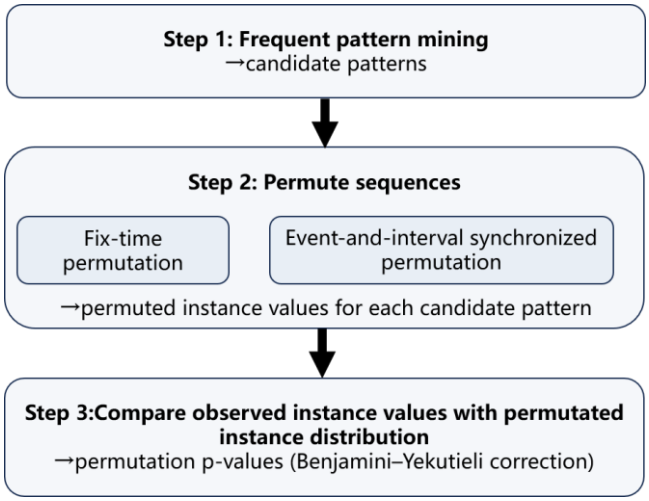

Figure 1. Workflow of the permutation testing.

Table 1. Characteristics of datasets

Learning environment	Learners	Mean Sequence Length	Event types
Aqualab	590 Middle School Students	817	58
LiRuYun	68 Undergraduate	549	27
iSnap	54 Undergraduates	2,896	21

3.2 Analyses

For each dataset, we first applied cSPADE[10] to the original data to mine frequent temporal sequential patterns. For each mined pattern, we then computed its observed instance value, which served as the test statistic in the permutation test. Next, we applied the two permutation techniques described in Section 2 and compared the original frequent pattern set with the post-permutation statistically significant pattern set. Comparisons focused on three aspects: (1)

[1] Aqualab and iSnap are publicly available datasets, LiRuYun is a proprietary learning management system; event logs were obtained through the course platform with participants' informed consent, and all records were fully anonymized prior to analysis.

the number of patterns, (2) pattern length distribution, and (3) the frequency distribution of constituent event types.

Before permutation testing, we ran preliminary cSPADE analyses with multiple parameter combinations for each dataset and selected settings that balanced candidate-pattern coverage and computational feasibility of the permutation testing. For Aqualab, the final settings were minimum support = 0.7, max gap = 2, and max time gap = 217s. For LiRuYun, the final settings were minimum support = 0.2, max gap = 1, and max time gap = 52s. For iSnap, the final settings were minimum support = 0.3, max gap = 1, and max time gap = 573s. The three time gaps were the 60th, 60th and 20th percentile of the pooled time intervals from the first to subsequent events within sequences in the corresponding datasets.

4. RESULTS AND DISCUSSION

4.1 RQ1: How many frequent patterns remained statistically significant after permutation testing?

Figure 2 shows the number of various patterns across datasets. On Aqualab, cSPADE yielded 435 candidate patterns; 390 remained significant under fixed-time permutation and 391 under event-and-interval synchronized permutation, corresponding to reductions of 10.3% and 10.1%. On LiRuYun, 374 candidates were mined; 286 and 315 remained significant under the two techniques, corresponding to reductions of 23.5% and 15.8%. On iSnap, 358 candidates were mined; 325 and 326 remained significant, corresponding to reductions of 9.2% and 8.9%.

These results suggest that frequent temporal pattern mining likely generates some patterns that do not survive significance testing. The proposed procedure therefore functions as a post-mining filter by removing patterns whose instance values are not significantly greater than expected under the conditions where learning events are randomly arranged. This matters because it moves learning event pattern evaluation beyond support alone and incorporates both temporal information and how strongly a pattern repeats in learning process.

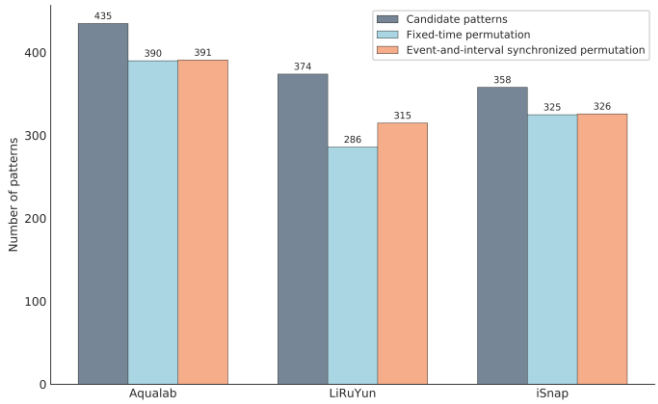

Figure 2. Comparison of pattern counts across datasets.

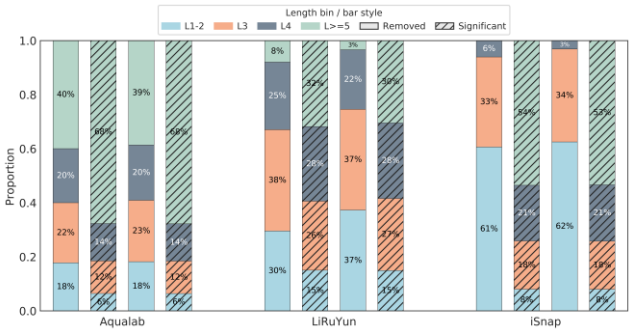

Figure 3. Proportions across length bins of removed and significant patterns.

Note. Within each dataset, the left two bars were the results using the fixed-time permutation, while the right two bars were the event-interval synchronized permutation.

4.2 RQ2: Whether significant and removed patterns differ in length?

Figure 3 shows a consistent pattern across all three datasets and both permutation techniques: longer patterns were more likely to remain statistically significant, whereas shorter patterns were more likely to be removed. In Aqualab, long patterns (length ≥ 5) made up the largest share of the significant set under both permutation methods, although they were also present among removed patterns. In LiRuYun and iSnap, the contrast was clearer: long patterns dominated the significant set, while removed patterns were concentrated in the shorter bins, especially L1–2 and L3. In iSnap, almost no long patterns were removed under either permutation technique. Chi-square tests indicated that the length distributions of removed and significant patterns differed statistically significantly in all datasets and both permutation techniques (χ^2 = 16.97~93.11, $p < 0.001$ in all conditions, and Cramér's V = 0.20 ~ 0.51). These results suggest that the permutation testing did not remove patterns uniformly across length bins. Instead, in these datasets, it more often filtered out shorter temporal patterns and retained longer patterns.

Meanwhile, pattern length alone should still be interpreted cautiously. Short patterns were not always noise, and long patterns were not always robust. For example, the proportion of significant patterns with length ⩽3 was substantial in LiRuYun (41% ~ 42%). This challenges the simplistic assumption that shorter patterns represent noises while longer patterns indicate meaningful results. In practice, learning environments can generate very different behavioral structures. Some environments support sustained inquiry or problem solving, in which learners engage in extended chains of actions while developing, revising, and testing ideas. Some of the others are more navigation-driven, where learners move through content in shorter and intermittent sequences, such as opening pages, checking modules, or switching between interface elements. As a result, the typical length of meaningful patterns may vary across learning platforms and tasks. Interpreting patterns based on length alone therefore risks misclassifying short but meaningful sequences as noise, or treating long sequences as inherently important when they may simply reflect the structure of the environment.

4.3 RQ3: Whether significant and removed patterns differ in the frequencies of their constituent event types?

For each dataset, we computed each event type's marginal frequency in the entire dataset and computed, for each candidate pattern, the mean of those frequencies across its constituent events. We then compared removed versus statistically significant patterns using Welch's t-tests. Table 2 reports the descriptive statistics and test results. In Aqualab and LiRuYun, the frequencies of constituent event-type of removed patterns were higher than significant patterns, and these differences were statistically significantly. In iSnap, the differences were in the same direction, although it was not statistically significant.

Table 2. Significant versus removed patterns by dataset.

Dataset	Patterns	*n*	*M (SD)*	*t*	*p*	Cohen's *d*
	Significant	390	0.24 (0.09)	4.82	0.00	0.59
Aqualab	removed	45	0.28 (0.05)			
LiRuYun	Significant	286	0.16 (0.09)	8.82	0.00	1.04
	removed	88	0.26 (0.08)			
iSnap	Significant	325	0.14 (0.03)	0.76	0.45	0.15
	removed	33	0.15 (0.04)			

These results suggest that some high-support patterns may be assembled from very frequent event types without reflecting reliable learning behavior structure. Permutation-based screening helps separate patterns that are easily assembled from frequent events from those that are more likely to reflect reliable behavioral structure. This distinction is illustrated by the example pattern pairs below.

Table 3. Patterns with similar support but different significance.

Dataset	Example patterns	Significant	Support	Mean observed instance value (mean permuted instance value)
iSnap	*Block.grabbed→Block.snapped→IDE.changeCategory→Block.created→Block.grabbed*	Yes	0.70	5.60 (0.05)
	Block.snapped→Block.clickRun	No	0.70	5.67 (5.70)
LiRuYun	*View the course module→View the course module→View the course module→View the course module*	Yes	0.72	9.96 (4.73)
	View the course→View the course→View the course module	No	0.75	1.79 (3.77)

4.4 Example patterns with similar support but different significance

To illustrate the difference between support and statistical significance, we selected two pairs of patterns in which one pattern was significant and the other was removed despite similar support. More specifically, for each pattern we simultaneously report its support, the mean of the observed instance values, and the mean instance value under permutation (in parentheses). If the observed level is not elevated relative to the permutation reference, then even a relatively high support may still be driven primarily by incidental assembly of highly frequent event types. Conversely, if the observed level is substantially higher than the permutation reference, this suggests that the pattern remains statistically salient under the corresponding null hypothesis. We illustrate these points using examples from LiRuYun and iSnap below. In LiRuYun, the removed pattern *View the course → View the course → View the course module* had support of 0.75, but its observed instance mean was lower than its permuted baseline (1.79 vs. 3.77). By contrast, the significant pattern *View the course module →View the course module → View the course module → View the course module* had comparable support (0.72) but a much higher observed instance mean than its permuted baseline (9.96 vs. 4.73). In terms of learning behaviors, the removed pattern might reflect incidental navigation, such as briefly checking course pages before opening a module. In contrast, the significant pattern might be more consistent with repeated engagement with the same module during a learning episode.

A similar contrast appeared in the block-based programming data. The significant pattern *Block.grabbed → Block.snapped → IDE.changeCategory → Block.created → Block.grabbed* and the removed pattern *Block.snapped →Block.clickRun* had nearly identical support values, yet only the former showed a large gap between observed and permuted instance values (5.60 vs. 0.05), whereas the latter did not (5.67 vs. 5.70). The removed pattern might reflect a routine snap-and-run action that occurs frequently but not greater than expected frequency by chance. In contrast, the significant pattern might be more consistent with sustained constructive work, where learners repeatedly organized, modified, and extended blocks within the same problem-solving episode.

These examples show that patterns with similar support can differ substantially in their statistically significance and behavioral meaning. The proposed permutation-based screening assists in removing statistically weak patterns and distinguishing patterns that are more likely to reflect meaningful learning behaviors from those that may primarily reflect routine navigation or interaction.

5. LIMITATIONS AND FUTURE WORK

This study has several limitations. First, large candidate pattern sets and repeated permutations make the testing procedure resource-intensive. Future work should improve efficiency with parallelization and adaptive stopping.. Second, current permutation techniques assume exchangeability of inter-event gaps, which may not hold for autocorrelated or non-stationary learning behaviors. Third, results were based on a small number of datasets and contexts. Replication across more platforms and links to learning outcomes would strengthen generalizability and educational relevance.

6. ACKNOWLEDGMENTS

This study was funded by National Natural Science Foundation of China (No.62407014).

7. REFERENCES

[1] Wong, J., Khalil, M., Baars, M., De Koning, B.B. and Paas, F. 2019. Exploring sequences of learner activities in relation to self-regulated learning in a massive open online course. *Computers & Education*. 140, (Oct. 2019), 103595. https://doi.org/10.1016/j.compedu.2019.103595.

[2] Zhang, Y. and Paquette, L. 2023. Sequential pattern mining in educational data: the application context, potential, strengths, and limitations. *Educational data science: essentials, approaches, and tendencies*. A. Peña-Ayala, ed. Springer Nature Singapore. 219–254.

[3] Jamshed, A., Mallick, B. and Bharti, R.K. 2024. A systematic review on sequential pattern mining-types, algorithms and applications. *Wireless Personal Communications*. 138, 4 (Oct. 2024), 2371–2405. https://doi.org/10.1007/s11277-024-11605-2.

[4] Jenkins, S., Walzer-Goldfeld, S. and Riondato, M. 2022. SPEck: mining statistically-significant sequential patterns efficiently with exact sampling. *Data Mining and Knowledge Discovery*. 36, 4 (July 2022), 1575–1599. https://doi.org/10.1007/s10618-022-00848-x.

[5] Zhang, Y., Paquette, L. and Bosch, N. 2025. Using Permutation Tests to Identify Statistically Sound and Nonredundant Sequential Patterns in Educational Event Sequences. *Journal of Educational and Behavioral Statistics*. 50, 3 (June 2025), 387–419. https://doi.org/10.3102/10769986241248772.

[6] Sun, J.C.-Y., Liu, Y., Lin, X. and Hu, X. 2023. Temporal learning analytics to explore traces of self-regulated learning behaviors and their associations with learning performance, cognitive load, and student engagement in an asynchronous online course. *Frontiers in Psychology*. 13, (Jan.2023),1096337. https://doi.org/10.3389/fpsyg.2022.1096337.

[7] Molenaar, I. and Wise, A.F. 2022. Temporal aspects of learning analytics - grounding analyses in concepts of time. *The handbook of learning analytics*. C. Lang, G. Siemens, and A.F. Wise, eds. SOLAR.

[8] Fournier-Viger, P., Li, Y., Nawaz, M.S. and He, Y. 2022. FastTIRP: efficient discovery of time-interval related patterns. *Big data analytics*. P.P. Roy, A. Agarwal, T. Li, P. Krishna Reddy, and R. Uday Kiran, eds. Springer Nature Switzerland. 185–199.

[9] Lai, F., Chen, G., Gan, W. and Sun, M. 2024. Mining frequent temporal duration-based patterns on time interval sequential database. *Information Sciences*. 665, (Apr. 2024), 120421. https://doi.org/10.1016/j.ins.2024.120421.

[10] Zaki, M. J. (2001). SPADE: An efficient algorithm for mining frequent sequences. *Machine Learning*, *42*(1–2), 31–60. https://doi.org/10.1023/A:1007652502315.