



HAL
open science

Mining Discriminative Sequential Patterns of Self-regulated Learners

Amine Boulahmel, Fahima Djelil, Jean-Marie Gilliot, Philippe Leray, Gregory Smits

► **To cite this version:**

Amine Boulahmel, Fahima Djelil, Jean-Marie Gilliot, Philippe Leray, Gregory Smits. Mining Discriminative Sequential Patterns of Self-regulated Learners. International Conference on Intelligent Tutoring Systems, Jun 2024, Thessalonique, Greece. pp.137-149. hal-04617383

HAL Id: hal-04617383

<https://hal.science/hal-04617383>

Submitted on 4 Jul 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Open licence - etalab

Mining Discriminative Sequential Patterns of Self-Regulated Learners

Amine Boulahmel¹[0000-0002-2390-7081], Fahima Djelil¹[0000-0001-8449-2062],
Jean-Marie Gilliot¹[0000-0003-4569-3552], Philippe Leray²[0000-0002-0207-9280],
and Gregory Smits¹[0000-0002-0436-9273]

- ¹ IMT Atlantique, Lab-STICC, UMR CNRS 6285, F-29238 Brest, France
{amine.boulahmel, fahima.djelil, jm.gilliot,
gregory.smits}@imt-atlantique.fr
- ² Nantes Université, CNRS, LS2N UMR 6004, F-44000 Nantes, France
philippe.leray@univ-nantes.fr

Abstract. This research explores the links between self-regulation behaviors and indicators of learning performance. A data mining approach coupled with appropriate qualitative measures is proposed to extract behavioral sequences that are representative of learning success. Applied on an online programming platform, obtained results allowed to highlight important self-regulation behaviors during the planning and engagement phases. It e.g. appears that successful self-regulated learners are those who analyze their tasks before working on them. This work brings methodological contributions in the field of self-regulation learning measurement and is a first step towards the design of intelligent tutoring systems.

Keywords: Self-Regulated Learning, Discriminative Sequential Patterns Mining, Learning Performance, Programming Tasks, Learning Analytics.

1 Introduction

Although Self-Regulated Learning (SRL) has a long existence in the education literature [17], it is still a key challenge in modern Online Learning Environments (OLE) [6]. The measurement of SRL has gained increased research interest aimed at a more refined understanding of SRL behaviors and strategies [7].

OLEs make it possible to capture and store students' interactions with the platform as trace data, on which fine-grained analyses can be applied to *in fine* provide learners with accurate feedback on their self-regulation skills.

In this paper, the relationship between SRL behaviors and learner performance is studied. It is assumed that learning sessions are discretized in pedagogical activities ending with a success or failure indicator of performance. The objective of this work is to discover discriminative behavioral sequential patterns that more likely lead to learning success.

The following main question is addressed: How to leverage trace data to point out the link between learners' SRL behavior and learning performance? Answering this question is a first step towards the conception of an Intelligent Tutoring

System (ITS) to support learning. Provided contributions are: 1) a data mining pipeline including appropriate measures to extract sequential patterns of actions that are characteristic of learning performance, and 2) an implementation on real data collected from an OLE, demonstrating the relevance of this approach.

The paper is structured as follows. Section 2 gives the principles of the proposed approach. Section 3 describes a positioning wrt. existing works in SRL measurement, from one side, and discriminative sequential pattern mining, from the other side. The proposed sequence mining method is detailed in Section 4, its implementation and data collection are described in Section 5. Results are discussed in Section 6, conclusions and implications are drawn in Section 7.

2 Principle and Notations

2.1 Assumptions

Our main objective is to quantify the relevance of sequences of action sequences wrt. learning performance.

Collected sequences of actions end up with an indicator of performance which is a success or a failure in an exercise. Sequences can thus be divided into two sets, those that lead to success (positive set) and oppositely those that lead to failure (negative set). The more a sequence of actions is observed in the positive set and the less in the negative set, the more characteristic of success is the sequence. A sequence is also all the more characteristic of success, in that its actions have to appear in a specific order.

Moreover, this work relies on the SRL model of Siadaty et al. [13] that suggests three common phases of learning behaviors, from planning through engagement, to self-reflection and evaluation. It maps trace data to micro-level SRL processes that are already categorized under macro-level processes. In our case, phases that can be observed from the data are planning and engagement. Reflection and evaluation are not instrumented in the concerned OLE. Planning through *task analysis* includes action events such as clicking on exercise assignments and documentation. Engagement through *working on the task* and *applying strategy changes*, include respectively action events such as keystrokes, click-streams, mouse movements, and reviewing activities.

Finally, the extent to which a sequence is characteristic of success depends on its **discrimination degree** with the set of negative sequences, but also on the **order dependency of its actions**. Although the two phases of SRL planning and engagement can be observed and micro-level processes that affect learning performance can be examined from trace data, it is very likely that there is no unique behavior, reflected in a sequence of actions, that characterizes the performance of self-regulated learners.

Behavioral sequences must therefore be analyzed at different levels: platform, exercise, and learner. Defining the two sets of positive and negative sequences is crucial to the analysis.

2.2 Preliminaries

From trace data, sequences of SRL actions performed by each learner can be reconstructed. A set of task actions is denoted by \mathcal{A} , and a sequence, often denoted by s hereafter, is an ordered list of actions $s = \langle a_1, \dots, a_{|s|} \rangle$ where $|s|$ is the length of s and $a_i, i = 1..|s| \in \mathcal{A}$. A sequence s' is a sub-sequence of another sequence s , denoted by $s' \subseteq s$, if it exists a one-to-one mapping from s' to s that preserves the order of s and s' . One denotes by $supp(s, \mathcal{S})$ the support of the sequence s in a set \mathcal{S} of sequences and is computed as follows:

$$supp(s, \mathcal{S}) = \frac{|\{s' \in \mathcal{S}, s \subseteq s'\}|}{|\mathcal{S}|} \quad (1)$$

Definition 1. *A sequence s is said to be **frequent** if $supp(s, \mathcal{S}) \geq \alpha$, α being a predefined frequency threshold.*

Sequences that are characteristic of learning success are identified through a differential analysis between two sets of sequences denoted \mathcal{S}^+ and \mathcal{S}^- that gather the sequences leading to success and failure respectively. In addition to being frequent, sequences of interest in a differential analysis have to be discriminative of the positive set \mathcal{S}^+ compared to the negative one \mathcal{S}^- , which means that the sequence has to cover a larger ratio of \mathcal{S}^+ than \mathcal{S}^- .

3 Related Works

3.1 Sequence Mining for SRL Measurement

The use of sequence mining to measure SRL has emerged as a promising method, as it appears in some recent works. It is mainly applied to identify SRL behaviors and student profiles, as for [14] to reveal learning sequences on trace coded behaviors and strategies, and to analyze behavior patterns and how they affect the course performance [4]. Describing students' SRL patterns is an important analysis task performed with sequence mining. For example, it allows to explore coded sequences of learning actions to investigate the effects of personalized scaffolding on students' learning activities [11]. Similarly, it enables the exploration of behavioral patterns of SRL and the examination of the significance of self-regulation scales among different groups of students [15], as well as the exploration of sequences of learners' activities to understand how learners utilize the SRL supports [16]. To our knowledge, no work has focused on the mining of discriminative sequential patterns in the measurement of SRL behaviors. In this work, we focus on the most relevant qualitative measures to identify sequential patterns expressing SRL behaviors that affect learners' performance.

3.2 Discriminative Sequential Pattern Mining

A key constraint that a sequential pattern must satisfy to be considered interesting is related to its frequency of occurrence in the analyzed dataset [9].

When quantifying the informativeness of a pattern in a differential setting materialized by two subsets of sequences, an additional criterion to its frequency is often considered. A pattern is indeed interesting if it is both frequent in its subset of assignment and discriminative wrt. other subsets. In [3], the quantification of the discriminative degree of a given sequence is studied in a general setting when multiple classes of sequences exist. But, in most cases, sequences are split into two sets and a sequence is all the more discriminative as it frequently occurs in its set and not in the other one.

A common way to quantify the extent to which a sequence s is characteristic of a set \mathcal{S}^+ wrt. another one \mathcal{S}^- is to calculate the Support Difference (SD) [12]:

$$SD(s) = \text{supp}(s, \mathcal{S}^+) - \text{supp}(s, \mathcal{S}^-) \quad (2)$$

Compared to other existing measures of discrimination (see [12] for more details), the SD measure (Eq. 2) ranges in $[-1, 1]$, thus making its output easier to interpret. Another important property to consider when providing users with the most discriminative sequences is to check that the discriminative degree of a sequence s is not due to any of its sub-sequences, which would lead to redundancy in the provided list of discriminative sequences [10].

4 Mining Successful Behavioral Sequences

The first step is to determine the properties, and subsequently the appropriate measures, of the sequences of behavioral actions that are characteristic of learning success. Candidate sequences to characterize a learning success are obviously taken from \mathcal{S}^+ and have first to be frequent, this set of frequent sequences is denoted by \mathcal{S}_F^+ and built as follows:

$$\mathcal{S}_F^+ = \{s \in \mathcal{S}^+, \text{ st. } \text{supp}(s, \mathcal{S}^+) \geq \alpha\} \quad (3)$$

where α is a predefined frequency threshold that can be adjusted to control the number of candidate sequences to take into account in the next steps. To be characteristic of success, a sequence also has to be differential from sequences that lead to failure. This is where the measure SD (Eq. 2) comes into play.

Definition 2. *A sequence s is **discriminative** of \mathcal{S}^+ wrt. \mathcal{S}^- if $SD(s) > 0$.*

A sequence s may be discriminative but also redundant, as s may contain a sub-sequence that is itself discriminative. For instance, let us consider a frequent and discriminative sequence $s \in \mathcal{S}_F^+ : \langle a_1, a_2, a_3 \rangle$ and one of its sub-sequences $s' : \langle a_1, a_2 \rangle$ that is necessarily frequent too but also more discriminative ($SD(s') > SD(s)$). Then, it is useless to present the user both s and s' as s' is by itself frequent and discriminative. The measure $\overline{red}(s, \mathcal{S}_F^+)$ quantifies the redundancy of a sequence s according to its discriminative sub-sequences:

$$\overline{red}(s) = SD(s) - \min(SD(s), \max_{s' \subset s} SD(s')). \quad (4)$$

The measure $\overline{red}(s)$ is a revision of the measure introduced in [10], so as to obtain a non-redundancy degree normalized in the unit interval. The sequence s being compared to its discriminative sub-sequences, then $SD(s) \in]0, 1]$ and $\max_{s' \subset s} SD(s') \in]0, 1]$.

The measures of discrimination and non-redundancy are classically used in existing approaches to extract discriminative sequential patterns. In order to identify sequences of SRL actions characterizing success, an additional property has to be considered. A sequence of actions constitutes an SRL behavior if the order in which the actions appear in the sequence is important. Indeed, in the SRL model of learning [13], planning behaviors precede engagement behaviors, otherwise, it is the set of actions that matters and not the sequence itself. Thus, we quantify the extent to which the order in which the actions appear in a frequent sequence is decisive. To do so an order dependence measure, denoted $ordDep(s)$, is introduced that compares the discriminative degree of s with its permutations. It is based on the entropy measure of data series [1] :

$$ordDep(s) = SD(s) - \min(SD(s), \max_{s' \in \mathfrak{s}(s)} SD(s')), \quad (5)$$

where $\mathfrak{s}(s)$ denotes the set of permutations of s .

Definition 3. *A sequence expresses an SRL behavior that is characteristic of the learner's performance iff. it is both **frequent, discriminative, not redundant** and contains **order-dependent actions**. The overall qualitative degree attached to each sequence s to express the extent to which it constitutes an SRL behavior of \mathcal{S}^+ is denoted by $\mu(s)$ and computed as follows:*

$$\mu(s) = \min(SD(s), \overline{red}(s), ordDep(s)). \quad (6)$$

Lemma 1. *Based on the discriminative degree computed using SD , sequences having a strictly positive μ degree are the most characteristic sequences of \mathcal{S}^+ .*

Proof. It is straightforward to show that a sequence s having a strictly positive $\mu(s)$ is a representative sequence of \mathcal{S}^+ . As $s \in \mathcal{S}_F^+$ then s is frequent. Let $s' \subseteq s$ be a frequent sub-sequence (resp. permutation) of s . If s' is more discriminative than s then $\min(SD(s), SD(s')) = SD(s)$ and $\overline{red}(s) = 0$ (resp. $ordDep(s) = 0$) leading to $\mu(s) = 0$.

In summary, the first step of our approach leverages a sequential pattern mining algorithm [8] to extract all the patterns in \mathcal{S}^+ that occur frequently. Then, to keep discriminative patterns only, the support of the patterns that frequently occur in \mathcal{S}^+ has to be computed on \mathcal{S}^- . Moreover, using only the set of somewhat discriminative patterns (i.e. $SD > 0$), redundant and non order-dependant sequences are discarded. Especially to check the non-redundancy with their sub-sequences, discriminative sequences are processed in an increasing order of the size.

Table 1. Learning actions library for Quick-Pi trace data.

Learning Action	Code	Description
READ_TASK	RDT	The learner reads the task set by the exercise
NAVIGATION	NAV	The learner navigates through platform exercises
PROGRAMMING	PRG	The learner is programming, therefore attempting to solve the exercise
SUBMISSION_FAIL	SFL	The learner submits a code but the solution is invalid
SUBMISSION_SUCCESS	SSC	The learner submits a code and passes
CODE_DEBUG	CDG	The learner is debugging a code
CODE_TEST	CDT	The learner experiments with code before submitting
HOVER_TASK	HTK	The learner hovers over the task panel

5 Implementation and Data Collection

As an implementation of the proposed approach, a study on real data is conducted addressing the following questions: 1) What are the SRL behaviors that impact learner performance? 2) Is it meaningful to search for these behaviors at a *holistic* platform level or at a more individual level, i.e. learning task or learner level?

Data are collected from the platform Quick-Pi³, that provides learners with pedagogical content for learning programming along with a range of activities related to the Internet of Things for school or home use. The recorded data consists of timestamped raw traces describing the low-level interactions of the learner with the platform interface (i.e. clickstream data). The data are first cleaned: 1) Noisy captures correction, to correct noisy events captured by the platform, 2) Event overlap correction, to correct the time-stamped sampling errors, and 3) Learning actions fusion, to merge successive chunks of the same learning actions into single blocks. Then, raw traces are translated into interpretable learning actions. Events on the use of the platform are aggregated into interpreted learning actions leading to a learning action library (Table 1).

The participants are a total of $N = 506$ learners who connected online to the platform, within a period of 40 days. Data privacy is ensured as the platform does not collect any demographic or personal information about learners. All 8 exercises of the first module of Quick-Pi were attempted by the learners.

The data show that the number of attempters and the number of successful attempters tend to decrease as the difficulty of the exercise increases.

The dataset has been segmented into two sets, one containing sequential learning actions leading to a success \mathcal{S}^+ (sequences ending with SUBMISSION_SUCCESS), and the other containing sequential learning actions leading to failure \mathcal{S}^- (sequences ending with SUBMISSION_FAIL).

At the platform level, all the observed data sequences are split into two groups ($|\mathcal{S}^+| = 2139$ and $|\mathcal{S}^-| = 12927$). At the exercise level (24 exercises available),

³ <https://quick-pi.org/>

Table 2. Statistics on the frequent sequences computed at platform, exercise, and learner levels using SD , \overline{red} , $ordDep$ and μ .

	Platform				Exercise				Learner			
	min	max	avg	std	min	max	avg	std	min	max	avg	std
SD	-0.31	0.006	-0.10	0.09	-0.38	0.008	-0.23	0.03	-0.2	0.05	-0.10	0.03
\overline{red}	-0.10	0.06	-0.0004	0.02	-0.27	0.03	-0.001	0.02	-0.2	0.02	0.0003	0.01
$ordDep$	-0.27	0.02	-0.002	0.009	-0.27	0.02	-0.002	0.02	-0.2	0.002	-0.001	0.01
μ	-0.31	0.06	-0.10	0.09	-0.41	-0.01	-0.23	0.03	-0.25	-0.04	-0.16	0.01

there are averages of 89.12 positive sequences and 538.62 negative sequences. At the learner level (506 learners), the average number of positive (resp. negative) sequences per learner is 4.22 (resp. 25.54). The prefixspan⁴ algorithm has been used to mine frequent patterns, as it has the advantage of reducing processing time and memory compared with other apriori-like algorithms.

6 Results and Discussion

The discriminative sequential patterns mining at the three levels, platform, exercise, and learner, revealed that there are more frequent patterns that are characteristic of failure than of success (negative values of min and $avg SD$, and very low values of $max SD$) (see table 2). This can be explained by the nature of the data collected from the Quick-Pi platform, which is accessible online to a very wide audience and used in contexts that are not only school-based but also outside the classroom. These data therefore do not necessarily come from teacher-supervised sessions where self-regulation is supported.

6.1 SRL behaviors that impact the learner performance

Platform level. Based on the discriminative power computed using SD , \overline{red} , $ordDep$, sequences having a strictly positive μ value are the two atomic sequences $READ_TASK$ ($SD = \overline{red} = ordDep = \mu = 0.69$), and $CODE_DEBUG$ ($SD = \overline{red} = ordDep = \mu = 0.08$). This suggests that $READ_TASK$ is the most prevalent action within the set \mathcal{S}^+ , and thus the most indicative of success across the platform. This indicates the importance of task analysis (refereed by reading the task) within the SRL planning phase for learner performance. Although $CODE_DEBUG$ has low discriminative power compared to $READ_TASK$, it appears to be important for success in the programming tasks. Debugging code allows errors to be identified and helps the code to run well. For learners, this is very characteristic of reviewing tasks during their SRL engagement with the programming tasks. This second result shows the importance of reviewing tasks for learner performance.

⁴ <https://pypi.org/project/prefixspan/>

Table 3. Discriminative sequential patterns at the exercise level ($SD > 0.1$).

Exercise Code	Difficulty	Sequential pattern	Length	SD
MEL	Medium	$PRG \rightarrow RDT \rightarrow (PRG \rightarrow HTK)^3 \rightarrow CDG$	9	0.33
	Hard	$(PRG \rightarrow HTK)^3 \rightarrow CDG$	7	0.23
INS	Easy	$RDT \rightarrow PRG \rightarrow HTK \rightarrow CDG$	4	0.22
	Medium	$PRG \rightarrow HTK \rightarrow CDG \rightarrow RDT$	4	0.40
	Hard	$(PRG \rightarrow HTK)^3 \rightarrow PRG \rightarrow RDT \rightarrow (PRG \rightarrow HTK)^2 \rightarrow PRG \rightarrow CDG$	14	0.26
AVT	Easy	$(HTK \rightarrow PRG)^2 \rightarrow RDT \rightarrow CDG$	6	0.36
	Medium	$(PRG \rightarrow HTK)^3 \rightarrow PRG \rightarrow CDG$	8	0.44
	Hard	$RDT \rightarrow PRG \rightarrow NAV \rightarrow PRG \rightarrow HTK \rightarrow PRG \rightarrow RDT \rightarrow NAV \rightarrow PRG \rightarrow (RDT \rightarrow PRG \rightarrow (HTK \rightarrow PRG)^n)^m \rightarrow CDG$	$9+m$ $(2+2m)$	0.16
SRV	Easy	$RDT \rightarrow PRG \rightarrow HTK \rightarrow (PRG \rightarrow RDT)^2 \rightarrow CDG$	8	0.47
	Medium	$RDT \rightarrow (PRG \rightarrow HTK)^2 \rightarrow CDG$	6	0.21
	Hard	$RDT \rightarrow CDG \rightarrow RDT$	3	0.23

Exercise level. A total of 45 sequences appear to be representative of success ($\mu(s) > 0$). Table 3 lists the most discriminative sequences (based on $SD > 0.1$ although $\overline{red} = ordDep = \mu = 0$). These most discriminative sequences begin with the read task action (RDT), implying for learners a conscientious effort to ensure alignment with the task instructions, which is consistent with subsequent success (Table 3). Some sequences begin with the programming action (PRG) followed by reading (RDT) or hovering (HTK) the task. These behaviors are characteristic of task analysis in the SRL planning phase. In addition, these sequences end with either code debugging (CDB) or occasional reading of the task (RDT). These results indicate that during learners' SRL engagement phase, they typically review their code at the end of the task to ensure that it works correctly. Moreover, many exercises exhibit a learning strategy where learners iteratively program and check the completion of the task, resulting in sequence loops $(PRG \rightarrow HTK)^k$, where $k \in \mathbb{N}^*$ is the loop length. For instance, for the hard version of the exercise *AVT*, an extended sequence with a nested loop was observed $s = (RDT \rightarrow PRG \rightarrow (HTK \rightarrow PRG)^n)^m$, where $n, m \in \mathbb{N}^*$. This suggests that learners consistently apply strategy changes by reviewing their tasks during their SRL engagement phase, which may lead to prolonged task completion. Indeed, the length of the pattern in this case indicates a greater effort in problem-solving due to the difficulty of the exercise. For the exercise *SRV*, no loops were found, regardless of the difficulty of its versions, due to the limited number of learners who successfully completed it.

Learner level. A total of 488 sequences based on $\mu > 0$ were found at the learner level (i.e. 18 learners and one sequence per learner). As shown in table

Table 4. Most discriminative sequential patterns characteristic of success at the learner level ($\mu > 0$).

Learner	Sequential pattern	Length	SD	\overline{red}	$ordDep$	μ
L1	$(PRG \rightarrow HTK)^2 \rightarrow (PRG \rightarrow HTK)^2 \rightarrow PRG$	9	0.79	0.03	0.005	0.005
L2	$(PRG \rightarrow HTK)^2 \rightarrow PRG \rightarrow RDT$	6	0.85	0.03	0.05	0.05
L3	$(HTK \rightarrow PRG)^4 \rightarrow RDT$	9	0.90	0.10	0.005	0.005
L4	$RDT \rightarrow (PRG \rightarrow HTK)^3$	8	0.85	0.02	0.003	0.003
L5	$RDT \rightarrow HTK$	2	0.55	0.19	0.01	0.01
L6	$PRG \rightarrow CDT$	2	0.38	0.04	0.015	0.015
L7	$PRG \rightarrow RDT$	2	0.2	0.05	0.05	0.05
L8	$CDG \rightarrow HTK \rightarrow PRG \rightarrow RDT$	4	0.92	0.12	0.005	0.005
L9	$CDG \rightarrow PRG$	2	0.25	0.29	0.05	0.05
L10	$CDG \rightarrow HTK$	2	0.43	0.46	0.015	0.015
L11	$CDG \rightarrow PRG \rightarrow RDT \rightarrow HTK \rightarrow PRG$	5	0.95	0.01	0.01	0.01
L12	$(RDT \rightarrow PRG)^2 \rightarrow RDT$	5	0.68	0.05	0.005	0.005
L13	$CDG \rightarrow (PRG \rightarrow HTK)^2 \rightarrow (PRG \rightarrow RDT)^2$ $\rightarrow CDG \rightarrow CDT \rightarrow (RDT \rightarrow PRG)^2$	17	0.91	0.07	0.07	0.07

4, most of the learner sequences exhibit loops, mainly combining the actions programming and hovering over the task (ie. $(PRG \rightarrow HTK)^k$) with $k \in \mathbb{N}$. This behavior is characteristic of reviewing tasks during the engagement SRL phase. Different learners' patterns were identified. For instance, learner L13 shows a reviewing task tendency where he/she proceeds through code debugging (CDG), programming and reading or hovering the task, code testing (CDT), and again task reading and programming. The appearance of a CDT preceded by $(RDT \rightarrow PRG)^2$ potentially displays a state where learners have tested their program before submission, and made adjustments before proceeding with a submission. Learners L1, L2, L4, L8, L11, L12 showed quiet similar behaviors. Moreover, we observe that some sequential patterns start with code debugging. We can assume that these patterns occurred when learners had already attempted the tasks and possibly failed, before reviewing these tasks and changing their strategies to succeed in their tasks. Few patterns include code testing which is only displayed at the learner level, showing that most learners do not test their code before submission, which may explain the imbalance of occurrences of successful vs. failure sequences, skewed in favor of failure.

6.2 Discriminative sequential patterns at the platform, task and learner level

The discriminative sequential patterns mining at the different levels, platform, exercise, and learner shows that the proportion of discriminative sequences based on $\mu > 0$ tends to increase as we move from a holistic to a more individual level (ie. exercise level and learner-level). At the platform level, only two atomic patterns were identified (reading the task and debugging the code). This highlights the importance of these actions for learner's success at a *holistic* level. At the

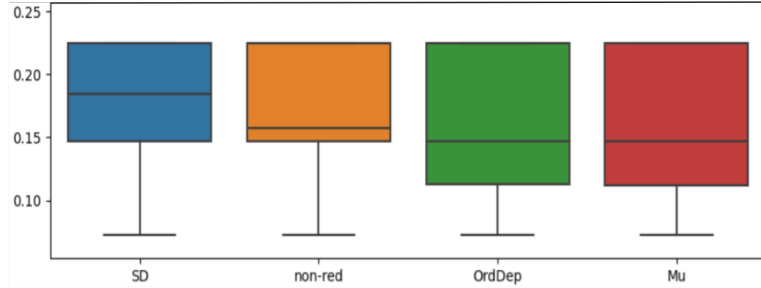


Fig. 1. Comparing the discriminative degree measures of sequences computed at the exercise level (SD , \overline{red} , $OrdDep$ and $\mu > 0$).

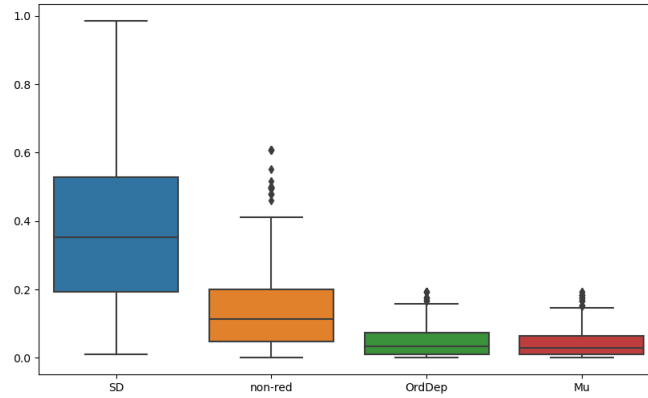


Fig. 2. Comparing the discriminative degree measures of sequences computed at the learner level (SD , \overline{red} , $ordDep$ and $\mu > 0$).

exercise level, very low variability of the discriminative measures values was observed (\overline{red} , $OrdDep$ and μ), see Figure 1. These measures have more variability at the learner level, as shown in Figure 2, where one can observe that these measures discriminate better at the individual level. This leads us to say that more discriminative behaviors may be observed at a user level than at a platform level.

7 Conclusion and Implications

In this work, we contribute to research in SRL with a new data mining approach that provides sequential patterns of SRL behaviors that may explain learning performance. A first contribution is the formalization of a generic data mining pipeline that includes appropriate measures for identifying self-regulated learner behaviors that lead to learning success. These measures allow us to determine the frequency, discriminative degree, non-redundancy, and order dependency of

sequential learner actions, leading to a more fine-grained analysis of SRL behaviors. A second contribution is the implementation of the proposed approach on real trace data from an online programming platform. The trace data is translated into learning actions, resulting in two sets of sequence data, on the one hand, sequences that lead to learning success, and on the other hand, sequences that lead to learning failure.

Results showed that self-regulated learners who demonstrated high performance were those who showed a planning SRL phase before an engagement phase. During the planning phase, the most successful learners were those who read their tasks before working on the programming tasks. During the SRL engagement phase, learners showed the behavior of reviewing tasks. This behavior appears to be essential for the completion of high-difficulty tasks, where learners constantly make strategy changes and invest more effort in problem-solving. We mine the data at three different levels, platform, exercise, and learner resulting with a different discriminative degree expressed by the proposed measures. This revealed that the more we analyze the patterns at an individual level, the more we identify the most characteristic behaviors impacting self-regulated learners' performance. This work has scholarly and practical implications. It provides interesting insights into the design of intelligent tutoring systems by providing behavioral actions to recommend as SRL scaffolds and strengthen learners' reflection. This work is not without limitations. Our approach uses only the success or failure of an exercise to explain the learning performance. It would be relevant to consider other modeling methods that estimate learner skill acquisition. Future work is motivated to design an approach that allows the exploration of behavioral patterns that contribute to learners' skill acquisition, based on a Bayesian Knowledge Tracing (BKT) model [2, 5]. Finally, it would be worth investigating how to provide learners with useful SRL scaffolds based on this data mining approach.

Acknowledgments. This work is part of the ANR xCALE project, funded by the French National Research Agency (ANR-20-CE38-0010).

References

1. C. Bandt and B. Pompe. Permutation entropy: a natural complexity measure for time series. *Physical review letters*, 88(17):174102, 2002.
2. A. Boulahmel, F. Djelil, J.-M. Gilliot, and G. Smits. Towards a skill-based self-regulated learning recommendation system. In *Doctoral Consortium of the 18th European Conference on Technology Enhanced Learning*, 2023.
3. H. Cheng, X. Yan, J. Han, and S. Y. Philip. Direct discriminative pattern mining for effective classification. In *2008 IEEE 24th International Conference on Data Engineering*, pages 169–178. IEEE, 2008.
4. A. Cicchinelli, E. Veas, A. Pardo, V. Pammer-Schindler, A. Fessler, C. Barreiros, and S. Lindstädt. Finding traces of self-regulated learning in activity streams. In *Proceedings of the 8th international conference on learning analytics and knowledge*, pages 191–200, 2018.

5. Q. Couland, P. Leray, and A. Boulahmel. Un modèle générique avec structuration des compétences et facteurs externes pour le bayesian knowledge tracing. In *Proceedings of the 11èmes Journée Francophones sur les Réseaux Bayésiens et les Modèles Graphiques Probabilistes*, 2023.
6. F. Djelil, J.-M. Gilliot, S. Garlatti, and P. Leray. Supporting self-regulation learning using a bayesian approach. some preliminary insights. In *International Joint Conference on Artificial Intelligence IJCAI-21, Workshop Artificial Intelligence for Education*, 2021.
7. Y. Fan, J. van der Graaf, J. Kilgour, M. Rakovic, J. Moore, I. Molenaar, M. Bannert, and D. Gasevic. Improving the measurement of self-regulated learning using multi-channel data. *Metacognition and Learning*, 17:1025–1055, 2022.
8. P. Fournier-Viger, J. C.-W. Lin, R. U. Kiran, Y. S. Koh, and R. Thomas. A survey of sequential pattern mining. *Data Science and Pattern Recognition*, 1(1):54–77, 2017.
9. J. Han. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2000.
10. Z. He, S. Zhang, F. Gu, and J. Wu. Mining conditional discriminative sequential patterns. *Information Sciences*, 478:524–539, 2019.
11. L. Lim, M. Bannert, J. van der Graaf, S. Singh, Y. Fan, S. Surendrannair, M. Rakovic, I. Molenaar, J. Moore, and D. Gasevic. Effects of real-time analytics-based personalized scaffolds on students’ self-regulated learning. *Computers in Human Behavior*, 139, 2023.
12. H. S. Pham, G. Virlet, D. Lavenier, and A. Termier. Statistically significant discriminative patterns searching. In *Big Data Analytics and Knowledge Discovery: 21st International Conference, DaWaK 2019, Linz, Austria, August 26–29, 2019, Proceedings 21*, pages 105–115. Springer, 2019.
13. M. Siadaty, D. Gasevic, and M. Hatala. Trace-based micro-analytic measurement of self-regulated learning processes. *Journal of Learning Analytics*, 3(1):183–214, 2016.
14. N. Srivasta, Y. Fan, M. Rakovic, S. Singh, J. Jovanovic, J. van der Graaf, L. Lim, S. Surendrannair, J. Kilgour, I. Molenaar, M. Bannert, J. Moore, and D. Gasevic. Effects of internal and external conditions on strategies of self-regulated learning: A learning analytics study. In *Proceedings of the 12th International Learning Analytics and Knowledge Conference*, pages 392–403, 2022.
15. J. C.-Y. Sun, H.-E. Tsai, and W. K. R. Cheng. Effects of integrating an open learner model with ai-enabled visualization on students’ self-regulation strategies usage and behavioral patterns in an online research ethics course. *Computers and Education: Artificial Intelligence*, 4, 2023.
16. J. Wong, M. Khalil, M. Baars, B. de Koning, and F. Paas. Exploring sequences of learner activities in relation to self-regulated learning in a massive open online course. *Computers & Education*, 140, 2019.
17. B. J. Zimmerman. A social cognitive view of self-regulated academic learning. *Journal of Educational Psychology*, 81(3):329–339, 09 1989.