

# Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques

Christoph Sonnenberg and Maria Bannert

University of Wuerzburg, Instructional Media, Germany

christoph.sonnenberg@uni-wuerzburg.de

**ABSTRACT:** According to research examining self-regulated learning (SRL), we regard individual regulation as a specific sequence of regulatory activities. Ideally, students perform various learning activities, such as analyzing, monitoring, and evaluating cognitive and motivational aspects during learning. Metacognitive prompts can foster SRL by inducing regulatory activities, which, in turn, improve the learning outcome. However, the specific effects of metacognitive support on the dynamic characteristics of SRL are not understood. Therefore, the aim of our study was to analyze the effects of metacognitive prompts on learning processes and outcomes during a computer-based learning task. Participants of the experimental group (EG,  $n=35$ ) were supported by metacognitive prompts, whereas participants of the control group (CG,  $n=35$ ) received no support. Data regarding learning processes were obtained by concurrent think-aloud protocols. The EG exhibited significantly more metacognitive learning events than did the CG. Furthermore, these regulatory activities correspond positively with learning outcomes. Process mining techniques were used to analyze sequential patterns. Our findings indicate differences in the process models of the EG and CG and demonstrate the added value of taking the order of learning activities into account by discovering regulatory patterns.

**KEYWORDS:** self-regulated learning, metacognitive prompting, process analysis, process mining, think-aloud data, HeuristicsMiner algorithm

## 1 INTRODUCTION

Recent research in the field of self-regulated learning (SRL) has moved to a process-orientated or event-based view to investigate how learning processes unfold over time and how scaffolds influence the dynamic nature of regulatory activities. Two recent special issues indicate the importance of investigating sequential and temporal patterns in learning processes and present new methodological contributions for the analysis of time and order in learning activities (Martin & Sherin, 2013; Molenaar & Järvelä, 2014). Technical advances allow the recording of learning-related behaviour on a very detailed level and largely unobtrusively for learners (e.g., Azevedo et al., 2013; Winne & Nesbit, 2009). As such, researchers have focused more on behavioural process data and less on measures of aptitude (Azevedo, 2009; Bannert, 2009; Veenman, van Hout-Wolters, & Afflerbach, 2006). When focusing on process data, differences among learners are explained on the event level with respect to regularities and patterns (Winne & Perry, 2000), allowing researchers to gain new insights into the process of learning.

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

Process analysis methods beyond the variable-centred *coding and counting* approach (Kapur, 2011) can provide valuable information on the specific effects of scaffolds (e.g., metacognitive prompts) and are able to inform researchers about how to optimize an applied supporting strategy further (e.g., Jeong et al., 2008; Johnson, Azevedo, & D’Mello, 2011; Molenaar & Chiu, 2014). Moreover, findings on the sequential and temporal structure of SRL processes can provide knowledge for the development of SRL theories on the micro-level (Molenaar & Järvelä, 2014).

Our approach applies the techniques of process mining (Trčka, Pechenizkiy, & van der Aalst, 2010) on process data obtained by concurrent think-aloud protocols (Ericsson & Simon, 1993). For example, we have compared process patterns of students with high versus low learning performance in a recent study (Bannert, Reimann, & Sonnenberg, 2014) and demonstrated that process mining techniques can reveal differences in the sequential patterns of regulatory processes. Now, we are investigating the effects of *metacognitive prompts* (Bannert, 2009) by means of an in-depth analysis using process mining techniques. An analysis of differences in the process models between students supported by metacognitive prompts and students without prompts can provide information on how to promote beneficial regulatory patterns and thereby improve learning.

The paper is structured as follows: First, we introduce research focusing on the support of SRL through metacognitive prompts. Second, we describe SRL models that emphasize the importance of different learning events and event patterns. Third, some of the foundations of analyzing learning processes with process mining are introduced. Fourth, we analyze process data from coded think-aloud protocols of an experimental study. In addition to the traditional frequency-based approach, the relative arrangement of learning activities is taken into account using process mining techniques. Finally, the results of these analyses are compared, and the effects of metacognitive support on the sequential structure of SRL processes are discussed.

## 2 THEORETICAL BACKGROUND

### 2.1 Metacognitive Support through Prompts

Current research in metacognition and SRL shows that learners often do not spontaneously use metacognitive skills during learning, which in turn leads to poorer learning outcomes (e.g., Azevedo, 2009; Bannert & Mengelkamp, 2013; Greene, Dellinger, Tüysüzoglu, & Costa, 2013; Winne & Hadwin, 2008; Zimmerman, 2008). The students’ awareness and control of their own manner of learning is important, especially in technology-enhanced and open-ended learning settings (Azevedo, 2005; Lin, 2001; Lin, Hmelo, Kinzer, & Secules, 1999). In most open-ended learning environments, it is constantly necessary to make decisions on what to do and where to go next and to evaluate the retrieved information with respect to current learning goals (Schnotz, 1998). Therefore, the general purpose of our research is to provide metacognitive support for hypermedia learning through metacognitive prompts.

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

*Instructional prompts* are scaffolds that induce and stimulate students' cognitive, metacognitive, and motivational activities during learning (Bannert, 2009). The underlying assumption is that students have already acquired these processes, but they do not recall or execute them spontaneously in a specific learning situation (*production deficit*; Veenman et al., 2006; Veenman, 2007). Metacognitive prompts aim at inducing regulatory activities such as orientation, goal specification, planning, monitoring and control, and evaluation strategies (Bannert, 2007; Veenman, 1993) by asking students to reflect upon, monitor, and control their own learning process.

Previous research has demonstrated beneficial effects from metacognitive prompting (e.g., Azevedo, Cromley, Moos, Greene, & Winters, 2011; Ge, 2013; Johnson et al., 2011; Lin & Lehman, 1999; Veenman, 1993; Winne & Hadwin, 2013). For example, Lin and Lehman (1999) prompted students to give reasons for their actions to increase the awareness of their own strategies by utilizing a pop-up window at certain times in a computer-based simulation environment (e.g., "What is your plan for solving the problem?"). Their findings showed significantly higher performance on contextually dissimilar problems (i.e., far transfer performance) for the students supported by prompts. Based on an analysis of think-aloud data, Johnson et al. (2011) showed that prompts given by a human tutor during learning in a hypermedia learning environment influenced the deployment of regulatory processes and temporal dependencies. Compared to a control group, the externally assisted condition also achieved a better learning outcome.

In previous experiments, we investigated the effects of different types of metacognitive prompts during hypermedia learning (Bannert & Mengelkamp, 2013; Bannert & Reimann, 2012). The prompts stimulated or even suggested appropriate metacognitive learning activities for university students during a hypermedia learning session lasting approximately 40 minutes. For example, in one of our experiments, students were prompted after each navigational step in a learning environment to verbalize the reasons why they had chosen the next step (so-called *reflection prompts*; Bannert, 2006). Overall, the findings confirm the positive effects of all investigated types of metacognitive prompts on transfer performance and the use of learning strategies during learning.

Our most recent work (Bannert, Sonnenberg, Mengelkamp, & Pieger, 2015) investigates the effects of a new type of metacognitive prompt (so-called *self-directed metacognitive prompts*) on navigation behaviour and learning outcomes. In summary, the findings show that such prompts enhance strategic navigation behaviour (i.e., students visited relevant webpages significantly more often and spent more time on them) and transfer performance (i.e., students performed better at applying knowledge of basic concepts to solve prototypical problems compared with a control group). In addition, learner characteristics (e.g., prior domain knowledge or verbal abilities) were obtained by questionnaires, but they had no effects as covariates in our analyses. The present study extends this contribution by focusing on the sequential analysis of coded think-aloud data obtained during learning.

Despite the findings about the general effectiveness of metacognitive prompts, the specific effects of prompts on learning processes remain unexplained. More precisely, a closer look at the effects of

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

prompts on the sequential and temporal structure of SRL processes is necessary (e.g., Jeong et al., 2008; Johnson et al., 2011). Understanding this process at the micro-level would allow researchers to better design metacognitive support. For example, regulatory patterns associated with successful learning but that could not be fostered by metacognitive prompts could be identified. Subsequently, the metacognitive support could be adapted by taking information about these patterns into consideration. Therefore, we focus on analyzing the sequential order of learning activities obtained by concurrent think-aloud protocols during learning.

## 2.2 Regulatory Patterns in SRL

Boekaerts (1997) describes SRL as a complex interaction of cognitive, metacognitive, and motivational regulatory components. With respect to assumptions in SRL models (e.g., Winne & Hadwin, 2008; Zimmerman, 2008), successful studying corresponds with an active performance of different regulatory activities during learning. These regulatory activities include employing orientation to obtain an overview of the learning task and resources, planning the course of learning, monitoring and controlling all learning steps, and evaluating the learning product. Research in SRL has confirmed that successful learning is associated with the active deployment of these regulatory activities (e.g., Azevedo, Guthrie, & Seibert, 2004; Bannert, 2009; Johnson et al., 2011; Moos & Azevedo, 2009).

Most SRL models share the common assumption of a time-ordered sequence of regulatory activities, although they do not imply a strict order (Azevedo, 2009). Usually, three cyclic phases of forethought, performance, and reflection (Zimmerman, 2000) are distinguished. The forethought phase comprises task analysis, goal setting, and strategic planning. During the performance phase, self-observations for adaptations (monitoring) and control strategies (self-instruction or time management) are deployed. Finally, the reflection phase includes self-judgments and self-reactions, which, in turn, can inform the next forethought phase. The COPES model (Winne & Hadwin, 2008) represents a more elaborate description of regulatory processes in terms of an information-processing model. Here, learning occurs in three phases, namely, task definition, goal setting and planning, and studying tactics, and a fourth optional phase, adaptations to metacognition. In addition, monitoring and control are crucial elements in the COPES model. Monitoring is used to detect differences between current conditions (e.g., learning progress) and standards (e.g., predefined learning goals), which, in turn, activates control processes to reduce discrepancies (e.g., engaging more intensively in a certain topic).

## 2.3 Microanalysis Using Process Mining Techniques

In a recent study (Bannert et al., 2014), we suggested process mining (PM) as a promising method in SRL research. PM allows researchers to describe and test process models of learning that incorporate an event-based view and that are at the high end of process granularity. These process models are able to represent the workflow of activities (van der Aalst, Weijters, & Maruster, 2004). Therefore, we argue that PM is adequate for investigating regulatory patterns based on process assumptions conceptualized in SRL research, as described in the previous section. For example, PM or data-mining techniques can

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

extract patterns by analyzing process data (e.g., think-aloud protocols or log files), and the resulting patterns can be compared to assumptions of SRL models (e.g., the assumption of a time-ordered sequence of regulatory activities in successful learning or the concept of dynamic and cyclic patterns). Therefore, the observed behaviour in process data could be aligned with SRL models.

PM is an approach that can be used in the context of *Educational Data Mining* (Romero, Ventura, Pechenizkiy, & Baker, 2010). In this context, PM represents student activities as a process model derived from their log traces while using a computer-based learning environment. In general, PM methods allow researchers to discover process models inductively from activity sequences stored in an event log, test process models through conformance checking with additional event data, or the extension of existing models (Trčka et al., 2010). Especially in the context of computer-supported learning research, PM techniques are increasingly used to study learning from a process-oriented perspective (Reimann & Yacef, 2013; Schoor & Bannert, 2012). For example, PM techniques can be applied to modelling sequences of learning activities that have been recorded in log files or coded think-aloud data.

By using PM techniques to discover process patterns in SRL activities, we assume that the present process data — comprising temporally ordered event sequences — is directed by one or more mental processes, with each set of processes corresponding to a process model. Hence, a process model represents a system of states and transitions that produced the sequence of learning events. Usually, the performance of this system is driven by a plan for action. In the context of SRL, this plan can be a learning strategy or an external resource provided to the learner (e.g., prompts). A process model is able to express a holistic view of a process by modelling a system comprising states and transitions rather than a process-as-sequence perspective (Reimann, 2009).

With respect to related approaches, hidden Markov models (e.g., Jeong et al., 2008) also allow for expressing the holistic nature of a process by taking into account the entire sample of behaviour. However, this approach uses time-consuming iterative procedures; generally, the researcher has to pre-define the appropriate number of states, and the interpretation of the output model is often difficult (van der Aalst, 2011). There are, however, approaches for automatically selecting the appropriate number of states using the Bayesian Information Criterion (e.g., Li & Biswas, 2002). Additionally, hidden Markov models, as well as simple transition graphs and other low-level models, represent a lower abstraction level than the PM notation language (e.g., inability to represent concurrency, which typically results in more complex models). Finally, PM techniques have the advantage of explicitly dealing with noise (i.e., exceptional or infrequent behaviour), which is necessary when analyzing real-life event traces. For these reasons, we recommend PM techniques for analyzing sequences of learning activities (see Bannert et al., 2014 for more information regarding the comparison to other process analysis methods).

## 2.4 Research Questions and Hypotheses

Metacognitive prompts ask students explicitly to reflect, monitor, and control their own learning

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

process. They focus students' attention on their own thoughts and on understanding the activities in which they are engaged during learning (e.g., Bannert, 2006; Hoffman & Spatariu, 2011). Hence, it is assumed that prompting students to monitor and evaluate their own manner of learning will allow them to activate their repertoire of metacognitive knowledge and learning strategies, which will consequently enhance their learning process and learning outcome. However, according to previous work and research on metacognitive prompting, the use of metacognitive prompts has to be explained and practiced in advance to guarantee an adequate application during learning (e.g., Bannert, 2007; Veenman, 2007). Based on the findings of studies investigating the effects of metacognitive prompts (e.g., Azevedo et al., 2004; Bannert, 2009), we expect that students supported by metacognitive prompts will engage in more regulatory activities, as obtained by coded think-aloud protocols. Moreover, scaffolded SRL processes should result in better learning performance; that is, a positive effect on learning outcomes mediated by improved regulatory behaviour. Whereas these two hypotheses are based on a variable-centred view of learning processes, we assume that an event-centred analysis that takes into account the relative arrangements of multiple learning activities can provide additional information about the sequential structure of the regulatory behaviour induced by the prompts (e.g., a sequence of orientation activities, searching for relevant information, cognitive processing, and evaluation of progress are typically executed). Therefore, the effectiveness of metacognitive prompts can be analyzed on a micro-level, and the results can be used to derive implications for the improvement of metacognitive support. In detail, the following research questions are addressed in the present study:

1. Does metacognitive prompting during learning influence SRL processes by engaging students in more metacognitive learning events?
2. Does the number of metacognitive learning events mediate the effect of metacognitive prompting on learning outcomes?
3. Which sequential patterns of SRL activities are induced by metacognitive prompting compared to a control group without support?

## 2.5 Process Mining Using the HeuristicsMiner Algorithm

To analyze the relative arrangement of learning activities, we employed the PM approach (Trčka et al., 2010). The basic idea of PM is to use an event log to generate a process model describing this log inductively (process discovery). Furthermore, theoretical models or empirically mined models can be compared to event logs (conformance checking), and existing models can be extended (model extension). Fluxicon Disco Version 1.7.2 (2014) software was used for data preparation. Next, the event log was imported into the ProM framework Version 5.2 (2008), and PM was conducted. The ProM framework comprises a variety of PM algorithms that can be assigned to the functions of discovery, model checking, or model extension. For our analysis, we used the HeuristicsMiner algorithm (Weijters, van der Aalst, & de Medeiros, 2006) for process discovery.

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

We selected the HeuristicsMiner algorithm based on a comparison of seven state-of-the-art process discovery algorithms on the dimensions of accuracy and comprehensibility, provided by de Weerdt, de Backer, Vanthienen, and Baesens (2012). Accuracy is defined as the capability of a sound capturing of behaviour in an event log, omitting over- and underfitting (i.e., a process model should balance between generality and precision). Comprehensibility comprises simplicity and structuredness of the resulting process models, and thereby determines the complexity and ease of interpretation of the output. For the first time, real-life event logs containing log data from different information systems were used for benchmarking PM algorithms. Among the seven algorithms, the HeuristicsMiner was the best technique for the real-life logs used and the authors conclude that “HeuristicsMiner seems the most appropriate and robust technique in a real-life context in terms of accuracy, comprehensibility, and scalability” (De Weerdt et al., 2012, p. 671). In the following, we explain the general principle and functionality of this algorithm in more detail.

## 2.6 General Principle of the HeuristicsMiner

The general principle of the HeuristicsMiner algorithm is to take into account the sequential order of events for mining a process model that represents the control flow of an event log (Weijters et al., 2006). The event log containing case IDs, time stamps, and activities represents the data input. Based on this input, the algorithm searches for causal dependencies between activities by computing a dependency graph that indicates the certainty of a relation between two activities (e.g., event *a* is followed by event *b* with a certainty of 0.90). Finally, a so-called *heuristic net* is generated as an output model that constitutes a visual representation of the dependencies among all activity classes in the event log. The resulting process model can be adjusted by setting thresholds for the inclusion of relations in the heuristic net (for more details on parameter settings, see below).

In addition, the HeuristicsMiner is based on two main assumptions. First, each non-initial activity has at least one other activity that triggers its performance, and each non-final activity is followed by at least one dependent activity. This assumption is used in the so-called *all activities connected heuristic* (Weijters et al., 2006). Second, the event log contains a representative sample of the observed behaviour, which usually contains a certain amount of noise, especially if traces of human behaviour are stored in the event log. For example, in our study, a perfect trace of verbal utterances for all performed learning steps is unlikely. Therefore, the event log contains noise caused, for example, by a missing learning step that was not uttered or by disagreement among the raters during the coding procedure. It must be noted that there is also noise in other types of data (e.g., log file data). Consequently, an analysis method is needed that can abstract from noise and that can concentrate on the main relations among learning activities. It is a specific feature of the HeuristicsMiner to be robust to noise in the data. This is the main reason for the appropriateness of applying this PM algorithm to our event log.

An additional advantage of the HeuristicsMiner algorithm is that the mined model (heuristic net) can be converted into a formal *petri net*. A petri net can be described as a bipartite directed graph with a finite set of places, a finite set of transitions, and two sets of directed arcs, from places to transitions and from

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

transitions to places (Reisig, 1985). Thus, the resulting process model can be used as input for other PM algorithms, and it can be utilized in subsequent analyses (e.g., conformance checking between the model and a new event log). In contrast, the output model of another promising process discovery algorithm within the ProM framework that we used in previous process analyses (Bannert et al., 2014; Schoor & Bannert, 2012), called the *Fuzzy Miner* (Günther & van der Aalst, 2007), cannot be converted into a petri net (De Weerd et al., 2012). Therefore, the HeuristicsMiner was the first choice for our present analysis.

## 2.7 Functionality and Application of the HeuristicsMiner

Considering its functionality, the HeuristicsMiner algorithm uses several parameters that guide the creation of a process model and that can be adjusted to set the level of abstraction from noise and low-frequency behaviour. First, a frequency-based metric is used to determine the degree of certainty of a relation between two events,  $A$  and  $B$ , based on an event log. The dependency values, ranging between  $-1$  and  $1$ , between all possible combinations of events are computed using the following formula (Weijters et al., 2006, p. 7):

$$A \Rightarrow_w B = \left( \frac{|a >_w b| - |b >_w a|}{|a >_w b| + |b >_w a| + 1} \right)$$

Based on an event log  $W$ , the certainty of a dependency relation between two events,  $A \Rightarrow_w B$ , is computed using the number of times event  $a$  is followed by event  $b$ , subtracted from the number of times event  $b$  is followed by event  $a$ , and divided by the number of occurrences of these two relations, plus 1. The number of correct ( $a$  follows  $b$ ) and incorrect ( $b$  follows  $a$ ) event sequences influences the dependency value by the  $+1$  in the denominator. For example, an event log containing only correct sequences ( $a$  is always followed by  $b$ , but never vice versa), but with a low frequency of five observations, results in a certainty of  $5/6 = 0.83$ , whereas in the case of a high frequency of 50 observations, the certainty of a dependency relation between  $a$  and  $b$  would be  $50/51 = 0.98$ .

Moreover, the computed dependency values are used to construct a heuristic net (i.e., the output model). However, not all dependency relations are kept in the process model. Instead, the HeuristicsMiner algorithm concentrates on the main causal dependencies and abstracts from noise and low-frequency behaviour. At first, the all activities connected heuristic is applied. Therefore, only the best candidates (with the highest  $A \Rightarrow_w B$  values) regarding the dependency values are kept in the output model. Second, three threshold parameters are used for the selection of further dependency relations. The *dependency threshold* determines the cut-off value for the inclusion of dependency relations in the output model. Furthermore, the *positive observation threshold* defines the minimum number of necessary observed sequences. Finally, the *relative to best threshold* determines that only additional dependency relations with a lower difference to the best candidate are included in the output model. We refer to Weijters et al. (2006) for more information about these threshold parameters.



(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

In our analysis, the threshold parameters were kept at their default values of dependency threshold = 0.9, positive observation behaviour = 10, and relative-to-best-threshold = 0.05. As explained above, these threshold parameters can be used to adjust the level of abstraction of the output model. For example, reducing the cutoff-values would result in additional dependency relations in the model and thus increase the complexity. However, there were no reasons for changing the default-values in our case.

Furthermore, the HeuristicsMiner algorithm can also address *short loops* of lengths one (e.g., ACCB) and two (e.g., ACDCDB) as well as *long distance dependencies*; that is, a dependency based on choices made in other parts of the process model. Moreover, the algorithm considers *AND-relations* (two events are executed concurrently) and *OR-Relations* (e.g., either event *b* or event *c* can be executed after event *a*) to construct the heuristic net.

In general, searching for an optimal process model based on a present event log can be challenging, especially if there is a certain amount of noise and less-frequent behaviour in the data. Therefore, it is possible to compare the resulting process model with the event log using a *fitness* value (Rozinat & van der Aalst, 2008). The fitness indicates the gap between the observed behaviour, that is, the set of event sequences in the log, and the mined process model.

By applying the HeuristicsMiner algorithm to our event log, we assume that the present set of sequences of learning events is caused by one or multiple underlying processes. However, it might be possible that there is a high variety in SRL activities within the sample. In this case, using very robust algorithms such as the HeuristicsMiner can result in over-generalization (underfitting); that is, the mined model allows for much more behaviour than what is actually observed (De Medeiros et al., 2008). Therefore, the event log could be modelled more precisely by generating different process models for subsets of participants instead of a single model for all cases. This approach is called *trace clustering*, which can improve the discovery of process models (De Weerd, vanden Broucke, Vanthienen, & Baesens, 2013; Greco, Guzzo, Pontieri, & Saccà, 2006). A plug-in has been implemented in the ProM framework that combines the HeuristicsMiner algorithm with a trace clustering procedure, namely, *DWS mining* (Disjunctive Workflow Schema; De Medeiros et al., 2008). The basic idea of DWS mining is to split the log into clusters iteratively until the mined process model for each cluster reaches high precision. A process model has a high precision if it only allows for behaviour that was observed in the event log. Consequently, a cluster is further partitioned if the mined model allows for more behaviour than is expressed by the cases within this cluster. For more information on the DWS mining plugin, refer to De Medeiros et al. (2008). In our analysis, we kept the default parameter settings for clustering the log traces.

### 3 METHOD

The present study extends a previous contribution (Bannert et al., 2015) that investigates the effects of metacognitive prompting on navigation behaviour and learning outcome referring to the same

participants, but to different research questions and to mostly different data.

### 3.1 Sample and Research Design

A total of n=70 undergraduate students from a German university participated in the study (mean age = 20.07, SD = 1.88, 82.9% female). All participants were either majoring in media communications or in human–computer systems. Participants were recruited via an online recruitment system administered by our institute, and each student received 40 Euros (approximately \$47 USD) for participating.

Altogether, the experimental study was based on a between-subject design and comprised two sessions. In the first session, learner characteristics were obtained as potential covariates (e.g., prior domain knowledge), especially in the case of an unbalanced distribution of characteristics among the groups by randomization (which is possible for the relatively small sample size). Approximately one week later, the participants were randomly assigned to either the experimental group (n=35) or the control group (n=35) and individually participated in hypermedia learning. The experimental group learned with metacognitive prompts, whereas the control group learned without prompts. Figure 1 presents an overview of the research design.

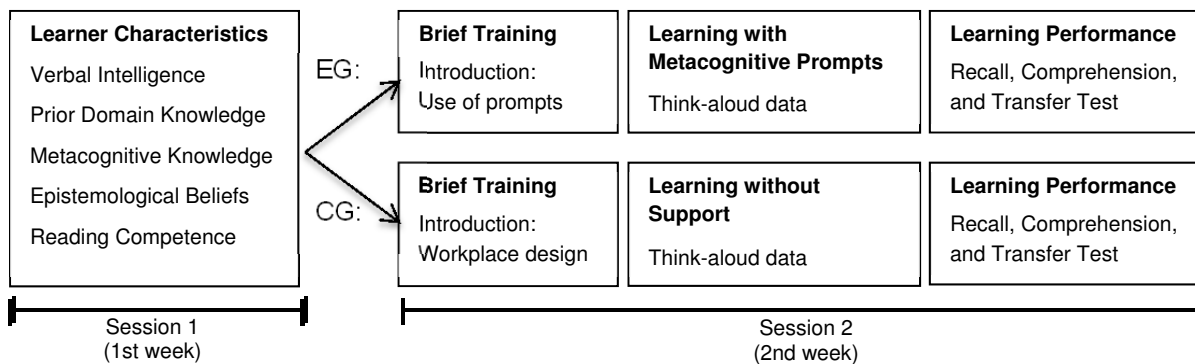


Figure 1. Research design

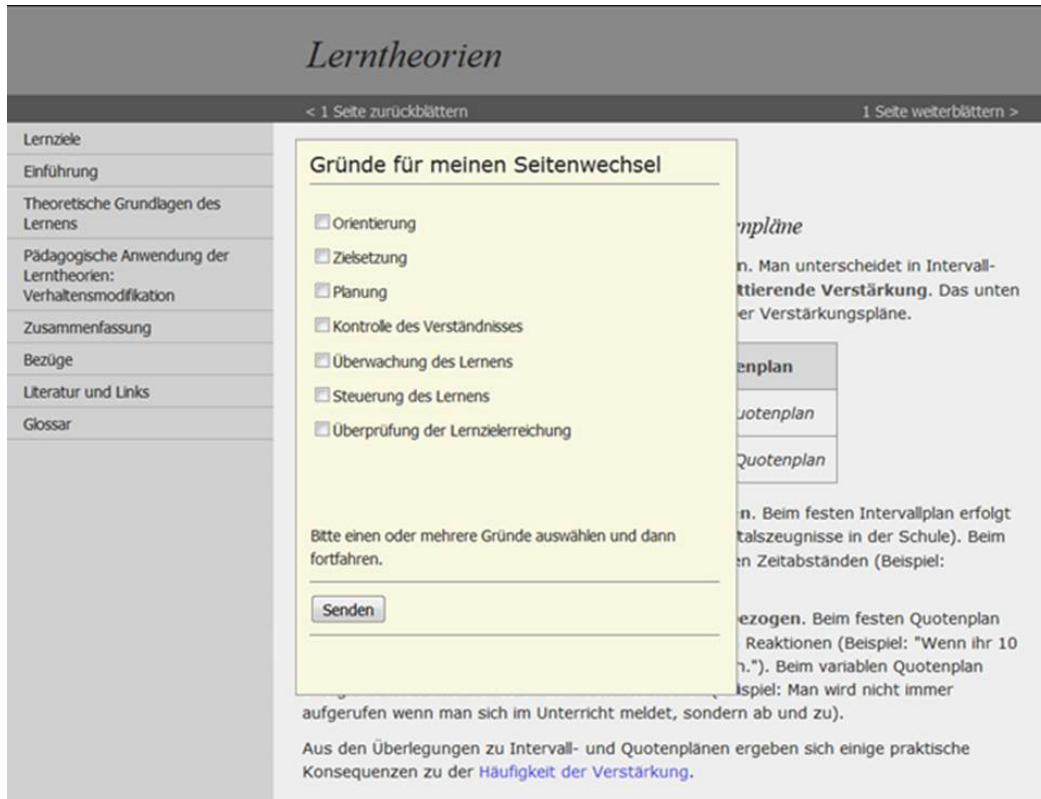
### 3.2 Learning Material and Performance Measurement

#### 3.2.1 Learning Environment and Metacognitive Prompts

The learning material comprised a chapter on the topic of learning theories (classical conditioning, operant conditioning, and observational learning) presented in a hypermedia learning environment. For example, the content of one node included a description of the Skinner-box with reference to the concept of operant conditioning, and illustrated with a picture. In total, the material comprised 50 nodes with 13,000 words, 20 pictures and tables, and 300 hyperlinks. Within this chapter, the material relevant for the learning task comprised 10 nodes with 2,300 words, 5 pictures and tables, and 60 hyperlinks. The remaining pages were not relevant for the learning task. These pages included overviews, summaries, and pages with information on concepts not relevant for the learning goals. The Flesch-Kincaid grade-level score of the complete learning material was 19.01.

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

Navigation in the learning environment was possible in four different ways: 1) a hierarchical navigation menu, 2) a next-page and previous-page button on top of each page, 3) the backward- and forward-button of the browser, and 4) hyperlinks embedded in the text.



**Figure 2: Learning environment with metacognitive prompt. Students are asked to select one or more reasons for node selection in a hypermedia learning environment by choosing among a list of strategic reasons (e.g., orientation, goal specification, planning) eight times during learning**

Support via metacognitive prompts was implemented in the learning environment. A prompt appeared in the form of a pop-up window placed in the middle of the screen eight times during learning. Each prompt contained a list of strategic reasons for node selection. At least one reason had to be selected before continuing with learning. Figure 2 shows the hypermedia learning environment with a metacognitive prompt.

### 3.2.2 Knowledge Tests

Learning performance was measured with three knowledge tests on different levels based on Bloom’s taxonomy of cognitive learning (Bloom, 1956). The measurement comprised a free recall test, a comprehension test, and a transfer test. In the free recall test, students were instructed to write down all basic concepts they could remember. The comprehension test, which assessed knowledge of facts, comprised 22 multiple-choice items, each with one correct and three incorrect answers. Transfer was

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

measured by asking students to apply basic concepts and knowledge of facts to solve eight prototypical problems in educational settings that were not explicitly addressed in the learning material (maximum score: 40 points). For example, students were asked to explain how a teacher should behave in response to a described classroom discipline problem based on the principles of operant conditioning. The answers of the participants were rated on a researcher-developed rating scale by two research assistants (Cohen's Kappa = .84). In case of disagreement among the raters, one of the authors determined the final score. More information on the learning material and knowledge tests used in our prompting studies is provided by Bannert and Reimann (2012) and by Bannert and Mengelkamp (2013).

### 3.3 Procedure

Approximately one week before the learning session, the learner characteristics *verbal intelligence*, *prior domain knowledge*, *metacognitive strategy knowledge*, *epistemological beliefs*, and *reading competency* were measured by questionnaires. More information on the instruments is provided in Bannert et al. (2015).

The learning session started with an introduction phase. First, the navigation in the hypermedia learning environment was explained by the experimenter. Then, the participant was asked to practice all possible ways of navigating the learning environment by using a practice lesson. After that, a series of exercises had to be performed in the practice lesson using concurrent thinking aloud during the task. The experimenter provided feedback and, if necessary, additional exercises until the participant firmly mastered the think-aloud technique.

Subsequently, the students in the experimental group received an introduction (approximately 10 minutes) to the use of metacognitive prompts, which included a description of the importance of reflecting on one's own learning steps, an explanation of the reasons for strategic node selection listed in the prompts, and the correct use of the prompts. It is necessary to explain the use of metacognitive prompts to the students to guarantee adequate application during learning (e.g., Veenman, 2007). After that, they were instructed to configure the prompts by arranging the list with reasons for node selection and by defining eight time stamps when the prompts should be presented during learning. To keep the workload for both groups equivalent, participants in the control group received an introduction to workplace design, which is not relevant for the stimulation of metacognitive learning activities. Instead of prompt configuration, they were asked to arrange their workplace before learning. Both introductions were realized by the experimenter using a sheet of instructions and advice visible to the participant.

Following this, the learning phase started. All participants received a sheet with their learning task, which instructed them to learn the basic concepts of operant conditioning within 40 minutes. Moreover, they were provided with a list of seven example concepts that had to be learned (e.g., Skinner Box, Positive Reinforcement). Students in the experimental group received metacognitive support by prompts, whereas the control group learned without prompts. All participants were completely free to navigate in the learning environment and to use their learning strategies. During learning, notes could

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

be taken on a blank sheet of paper (e.g., for summarizing or structuring information), but the participants were not allowed to use their notes to work on the knowledge tests. The participants were instructed to read and think aloud during the whole learning phase as practiced before, and these activities were videotaped. If a participant stopped thinking aloud for more than five seconds, the experimenter reminded her or him by saying, “Please think aloud.”

Directly after learning, the students worked on the three knowledge tests described above. Overall, the duration of the session was approximately two hours.

### 3.4 Coding Scheme

A coding scheme based on our theoretical framework of self-regulated hypermedia learning (Bannert, 2007) was used for segmenting and coding the students’ verbal protocols. Our theoretical framework characterizes hypermedia learning into the major categories *Metacognition*, *Cognition*, and *Motivation*. In addition, it distinguishes several sub-categories within the categories Metacognition and Cognition, as further described below.

Table 1 presents the coding categories and provides descriptions and examples. The coding scheme comprises the main categories Metacognition, Cognition, Motivation, and Other. Metacognition includes the sub-categories *Orientation*, *Goal specification*, *Planning*, *Searching for information*, *Judgment of its relevance*, *Evaluating goal attainment*, and finally *Monitoring and regulation*. Cognition contains *Reading*, *Repeating information*, and deeper processing, that is, *Elaboration* and *Organization of information*. The main category of Motivation includes all positive and negative utterances on the task, the situation, or oneself. Finally, all task-irrelevant utterances, non-classifiable utterances, and the handling of the prompts for the experimental group were assigned to the category Other.

The coding was conducted based on the procedure presented by Chi (1997). Segmentation of the verbal protocols was based on meaning. A segment was assigned for every definable learning activity. Multiple or nested codes were not allowed. Four trained research assistants coded the verbal protocols of all 70 participants. A random sample of three participants from each of the experimental group and the control group was selected to compute the interrater’s reliability. The reliability, based on 1,385 segments, showed substantial agreement: Cohen’s Kappa = .78, which is seen as sufficient for the following analysis.

### 3.5 Analysis

An example of the coded data used for the process analysis is presented in Table 2. The data comprise three types of information: 1) a Case ID that clearly distinguishes the participants, 2) a time stamp that indicates the beginning of an event, and 3) a learning activity — that is, the assigned category of the coding scheme (CODE). Using this information, it is possible to compute not only the frequency of events but also to determine the relative arrangement of multiple events. For example, in the short section of

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

Table 2, MONITOR is the most frequent activity (3 occurrences). Furthermore, MONITOR is directly followed by READ twice and directly followed by ORGANIZATION once.

**Table 1. Coding scheme for analyzing students’ learning activities**

Code	Coding Category	Description and Examples
<b>Metacognition</b>		
ORIENT	Orientation	Task clarification, overview of material. <i>I will sketch the menu first.</i>
SETGOAL	Goal specification	Goal setting and sub-goaling <i>I have to learn the basic concepts of operant conditioning.</i>
PLAN	Planning	Planning how to proceed <i>First I will decide in which sequence I have to learn and which pages to read.</i>
SEARCH	Search	Searching for information <i>Where is the page with the information about plans of reinforcement?</i>
EVALUATE	Judgment	Judgments about the relevance of information <i>Skinner’s Vita is not relevant for my learning task.</i>
EVAL	Evaluation	Checking and evaluating <i>Did I process all the topics?</i>
MONITOR	Monitoring	Monitoring one’s own learning <i>Ah, now I understand the principle.</i>
<b>Cognition</b>		
READ	Reading	Reading out loud
REPEAT	Repeating	Repeating
ELABORATE	Elaboration	Deeper processing, <i>paraphrasing, connecting, inferring</i>
ORGANIZATION	Organization	Organization <i>drawing a map, writing down major concepts</i>
<b>Motivation</b>		
MOT	Motivation	Positive, negative, neutral motivational utterances regarding a task, person, or situation <i>The task is very interesting.</i>
<b>Other</b>		
REST	Other	Off-topic statements, comments on technique, not interpretable statements, pauses <i>May I make notes? The mouse doesn’t work well.</i>

**Table 2. Section of an event log**

<i>Case ID</i>	<i>Timestamp</i>	<i>Learning Activity (CODE)</i>
		...
Case 1	04:14	ORIENT
Case 1	04:19	MONITOR
Case 1	04:24	READ
Case 1	04:37	REPEAT
Case 1	04:43	ORIENT
Case 1	05:01	MONITOR
Case 1	05:03	READ
Case 1	06:51	MONITOR
Case 1	06:56	ORGANIZATION
Case 1	07:30	REPEAT
		...

In the first step of our analysis, we took the frequencies of coded learning activities into account (frequency analysis) and used the frequency of metacognitive events to examine the expected mediation effect on transfer performance (mediation analysis). Subsequently, the sequential order of the coded learning activities is analyzed using a PM algorithm to discover differences in the process models of the experimental and control groups (process mining). Therefore, our view on temporality corresponds to the relative arrangement of multiple events.

## 4 RESULTS

A preliminary analysis showed that the randomized assignment of participants resulted in two subsamples with similar learner characteristics. With the exception of one subscale of reading competency — namely, *text comprehension* (measured by ELVES; Richter & van Holt, 2005) — no significant differences were found. In the case of the subscale text comprehension, students in the control group scored significantly better than those in the experimental group ( $t(69) = 2.97, p = .004, d = 0.72$ ; two-tailed testing). In conclusion, this analysis indicates that the following results are not caused by unbalanced subsamples.

### 4.1 Frequency Analysis

Table 3 presents the descriptive and test statistics of all coded events for the experimental group and the control group. In addition to the minimum and maximum occurrence of each category, absolute frequencies, means, and standard deviations are listed. A total of 8,743 events were coded for the students in the experimental group, and a total of 8,087 events were coded for the students in the control group. For the experimental group, there were, on average, approximately 250 events coded in

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

40 minutes of learning time, with 116 metacognitive, 107 cognitive, 2 motivational, and 25 other utterances. Participants in the control group showed a mean of approximately 231 events, with 98 metacognitive, 106 cognitive, 3 motivational, and 24 other utterances.

A one-tailed t-test for independent samples showed that the experimental and control groups significantly differ in the number of metacognitive utterances ( $t(69) = 1.80, p = .038, d = 0.44$ ). As expected, students in the experimental group who had been supported through metacognitive prompts showed a higher number of metacognitive learning activities ( $M = 116.43, SD = 45.97$ ) than students in the control group without prompts ( $M = 98.49, SD = 36.72$ ). Moreover, both groups showed a similar number of utterances in the remaining main categories Cognition ( $M_{EG} = 107.43, SD_{EG} = 36.01; M_{CG} = 106.31, SD_{CG} = 44.65$ ), Motivation ( $M_{EG} = 2.06, SD_{EG} = 4.14; M_{CG} = 2.54, SD_{CG} = 4.40$ ), and Other ( $M_{EG} = 25.34, SD_{EG} = 15.08; M_{CG} = 23.71, SD_{CG} = 12.19$ ). For these three categories, the t-tests for independent samples were not significant.

Concerning the descriptive statistics of the subcategories of Metacognition, the experimental group showed more Monitoring ( $M_{EG} = 71.17, SD_{EG} = 37.62; M_{CG} = 58.00, SD_{CG} = 27.46$ ), Orientation ( $M_{EG} = 14.31, SD_{EG} = 7.23; M_{CG} = 11.14, SD_{CG} = 5.80$ ), Evaluation ( $M_{EG} = 3.63, SD_{EG} = 3.08; M_{CG} = 2.49, SD_{CG} = 3.03$ ), and Planning ( $M_{EG} = 1.74, SD_{EG} = 1.65; M_{CG} = 0.74, SD_{CG} = 1.09$ ) compared to the control group. In both groups, the highest frequency occurred for Monitoring, followed by Orientation, Searching, and Judgment, whereas Planning and Goal specification were rarely executed by students. As reported in Table 3, on the right side, differences between the experimental and control groups are significant for Orientation, Planning, and Monitoring.

Within the main category Cognition, participants of the control group showed more reading activities ( $M_{EG} = 40.66, SD_{EG} = 15.75; M_{CG} = 44.49, SD_{CG} = 18.19$ ) but less Elaboration ( $M_{EG} = 21.91, SD_{EG} = 12.92; M_{CG} = 18.40, SD_{CG} = 17.54$ ) and less Organization ( $M_{EG} = 26.37, SD_{EG} = 13.87; M_{CG} = 24.60, SD_{CG} = 12.51$ ) than participants of the experimental group. However, all differences regarding these categories are non-significant. Finally, motivational events seldom occurred in both groups and with non-significant differences.

## 4.2 Mediation Analysis

A mediation analysis was conducted to investigate whether the observed relationship between the treatment group and learning performance (outcome variable) is mediated by the number of metacognitive events during learning. Regarding measurements of learning outcome, only transfer performance (i.e., a post-test score) differed significantly between the experimental and control groups ( $M_{EG} = 20.61, SD_{EG} = 3.97; M_{CG} = 18.79, SD_{CG} = 4.30; t(69) = 1.85, p = .035, d = 0.45$ ; for more details on learning outcomes, see Bannert et al., 2015). Furthermore, both the number of metacognitive events and its sub-category Monitoring significantly correlate with transfer performance (Metacognitive events:  $r = .22, p = .033$ ; Monitoring:  $r = .32, p = .003$ ). Therefore, these two variables are regarded as possible mediators, and only transfer performance is included as an outcome variable.



**Table 3: Absolute frequencies, means, and test statistics of all coded learning events for the experimental group and the control group**

	Experimental Group (n=35)					Control Group (n=35)					<i>t</i>	<i>p</i>	<i>d</i>
	Min	Max	Absolute Frequency	<i>M</i>	<i>SD</i>	Min	Max	Absolute Frequency	<i>M</i>	<i>SD</i>			
<b>Metacognition</b>	34	242	4075	116.43	45.97	34	173	3447	98.49	36.72	1.804	.038	0.44
Orientation	5	30	501	14.31	7.23	2	28	390	11.14	5.80	2.025	.024	0.49
Planning	0	5	61	1.74	1.65	0	4	26	0.74	1.09	2.987	.002	0.73
Goal specification	0	10	72	2.06	2.36	0	8	67	1.91	1.84	0.282	.309	0.07
Search	1	32	414	11.83	7.54	1	57	453	12.94	10.64	-0.506	.308	-0.12
Judgment	2	23	409	11.69	5.70	0	33	394	11.26	7.35	0.273	.393	0.07
Evaluation	0	15	127	3.63	3.08	0	15	87	2.49	3.03	1.565	.061	0.38
Monitoring	11	203	2491	71.17	37.62	9	124	2030	58.00	27.46	1.673	.049	0.41
<b>Cognition</b>	47	201	3760	107.43	36.01	30	193	3721	106.31	44.65	0.115	.909	0.03
Reading	20	84	1423	40.66	15.75	19	89	1557	44.49	18.19	-0.941	.350	-0.03
Repeating	2	45	647	18.49	11.04	2	59	659	18.83	12.25	-0.123	.902	-0.03
Elaboration	3	55	767	21.91	12.92	0	56	644	18.40	17.54	0.955	.343	0.23
Organization	3	61	923	26.37	13.87	0	58	861	24.60	12.51	0.561	.576	0.14
<b>Motivation</b>	0	18	72	2.06	4.14	0	22	89	2.54	4.40	-0.476	.636	0.11
<b>Other</b>	8	69	887	25.34	15.08	6	58	830	23.71	12.19	0.497	.621	0.12
<b>Sum of all coded events</b>	126	473	8743	249.80	76.64	112	390	8087	231.06	76.27	1.023	.310	0.25

*Note:* Since we expected metacognitive prompting to increase the number of metacognitive utterances, we conducted one-tailed testing for metacognitive categories; elsewhere we conducted two-tailed testing;  $p < .05$ .

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

We used the PROCESS custom dialog box for SPSS based on the regression-based approach of Hayes (2013) to run the mediation analysis, which calculates bootstrapped confidence intervals (BCa CI) for the indirect effect and the measurement of effect size. There was a significant indirect effect of the treatment on transfer performance through the number of metacognitive events,  $b = 0.33$ , BCa CI  $[-0.01, 1.09]$ . Kappa-squared (Preacher & Kelley, 2011) was used to measure the effect size. The detected effect is relatively small,  $K^2 = .039$ , 95% BCa CI  $[.004, .127]$ . Furthermore, there was a significant indirect effect through the number of Monitoring events,  $b = 0.48$ , BCa CI  $[0.01, 1.24]$ . Again, this represents a small effect,  $K^2 = .058$ , 95% BCa CI  $[.006, .142]$ .

In summary, the number of metacognitive events and of its sub-category Monitoring could be identified as mediator variables. Metacognitive prompting increased the occurrence of metacognitive events, especially of Monitoring, which in turn enhanced the transfer performance. Due to the mediation effect of the sub-category Monitoring being even slightly larger than the effect of all metacognitive events, we conclude that the mediation is mainly driven by Monitoring. Figure 3 presents the mediation model, including Monitoring as mediator variable.

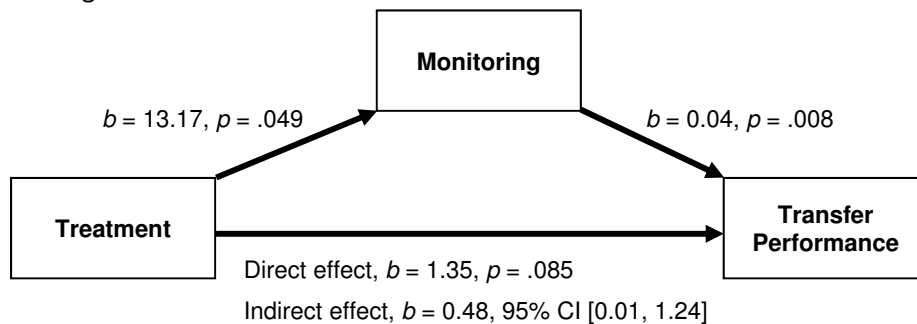


Figure 3: Mediation through the number of monitoring events

### 4.3 Process Analysis Using the HeuristicsMiner Algorithm

To apply the HeuristicsMiner, we had to simplify the categories of the coding scheme (see Table 1) for three reasons. First, the event classes Planning, Goal specification, and Motivation showed a very low frequency in our event log. Second, the HeuristicsMiner algorithm should preferably be used on data without too many different event classes (Rozinat, 2010). Finally, theoretical SRL models describe regulation processes mainly with the three phases of forethought, performance, and reflection (e.g., Zimmerman, 2000). The simplification was conducted as follows: We aggregated the metacognitive events Orientation, Planning, and Goal specification into a new event class called *Analyze*. Furthermore, the event class Judgment was added to Monitoring. In addition, the cognitive events Elaboration and Organization were combined to form a new event class called *Process*, that is, deeper processing. Finally, the event classes Motivation and Other were excluded from the process analysis. Altogether, seven event classes, listed in Table 4, were used for the analysis with PM techniques. With respect to the mean number of events, there was only a significant difference between both groups for the category Analyze ( $t(69) = 2.36, p = .011, d = 0.57$ ).

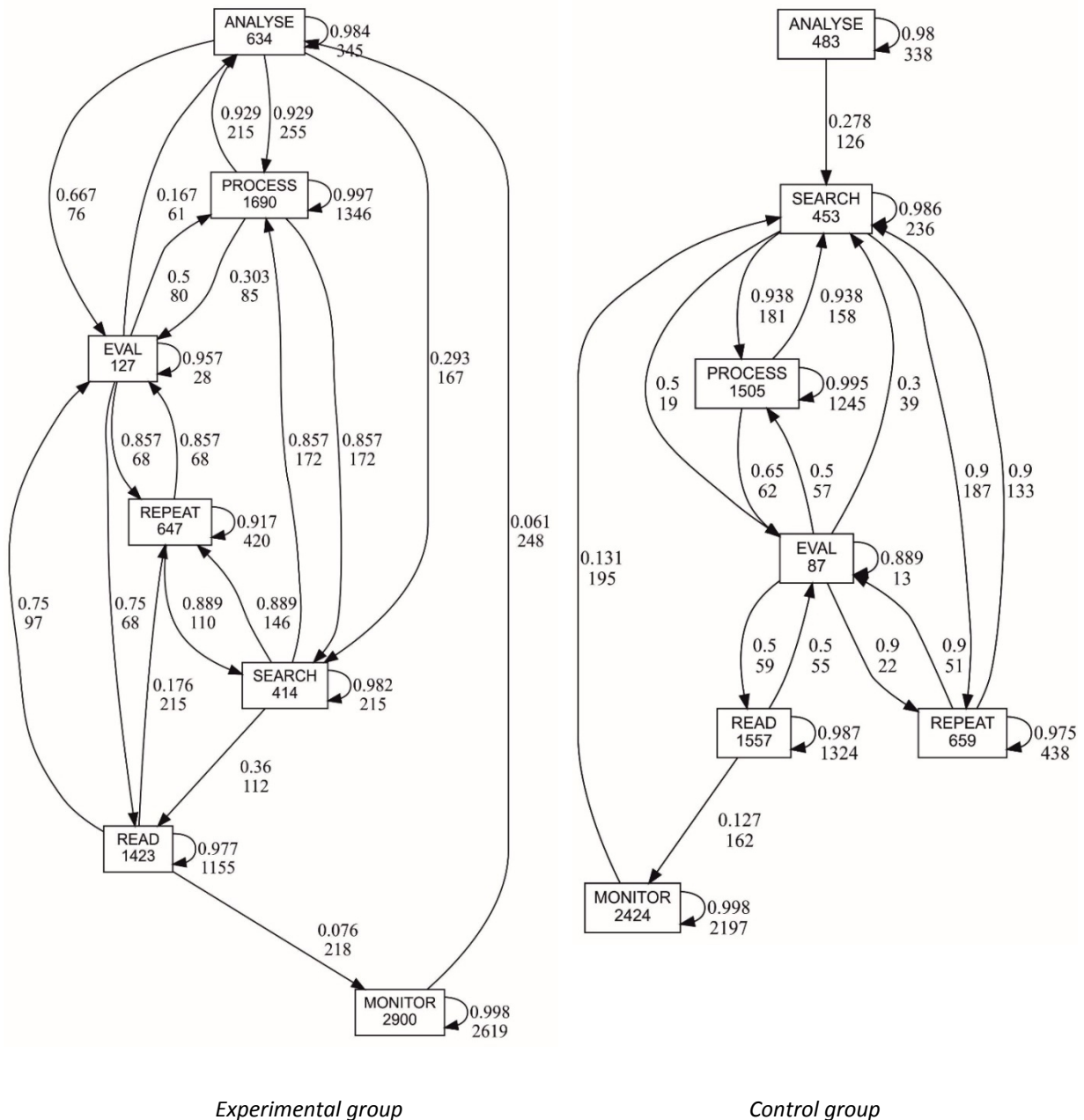
**Table 4: Absolute frequencies, means, and test statistics of aggregated categories for the experimental group and the control group**

	Experimental Group (n=35)					Control Group (n=35)					<i>t</i>	<i>p</i>	<i>d</i>
	Min	Max	Absolute Frequency	<i>M</i>	<i>SD</i>	Min	Max	Absolute Frequency	<i>M</i>	<i>SD</i>			
Analyze	5	37	634	18.11	8.58	3	33	483	13.80	6.61	2.356	.011	0.57
Search	1	32	414	11.83	7.54	1	57	453	12.94	10.64	-0.506	.310	-0.12
Evaluation	0	15	127	3.63	3.08	0	15	87	2.49	3.03	1.565	.061	0.38
Monitoring	20	211	2900	82.86	39.37	12	131	2424	69.26	30.71	1.612	.056	0.39
Reading	20	84	1423	40.66	15.75	19	89	1557	44.49	18.19	-0.941	.350	-0.03
Repeating	2	45	647	18.49	11.04	2	59	659	18.83	12.25	-0.123	.902	-0.03
Process	9	93	1690	48.29	21.54	3	84	1505	43.00	20.73	1.046	.299	0.25

*Note:* Since we expected metacognitive prompting to increase the number of metacognitive utterances, we conducted one-tailed testing for metacognitive categories; elsewhere we conducted two-tailed testing;  $p < .05$ .

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

The relative arrangement of learning activities was analyzed by applying the HeuristicsMiner algorithm in combination with the DWS mining plugin. The trace clustering did not split the cases into clusters for the participants of the experimental group (n=35) or the control group (n=35). This means a single process model can already express the event log with sufficient precision for both groups.



**Figure 4. Process models for the experimental group (n=35) and for the control group (n=35) represented as a heuristic net. Metacognitive Activities: ANALYZE = Orientation, Planning, and Goal specification; EVAL = Evaluation; MONITOR = Monitoring and Judgment. Cognitive Activities: READ = Reading; REPEAT = Repeating; PROCESS = Elaborate and Organization**

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

The resulting process models are displayed in Figure 4, where they are represented as heuristic nets for the experimental and control groups. These visual representations of the process models comprise square boxes that represent the event classes and arcs between these boxes that indicate the dependency between two event classes. The number in the event box represents the occurrence of an event class in the log. The arcs are labelled with two types of information. The upper number displays the dependency measure, which indicates the certainty of a dependency relation between two activities. A value close to 1.0 indicates a high certainty that a dependency relation exists. The lower number shows the number of times this transition is used, that is, how often event *a* is followed by event *b*. An arc pointing back at the same box indicates a self-loop, meaning that an event class often occurred multiple times in a row in loops of length one or length two (e.g., ACCB, ACDCDB).

The fitness between the mined model and the event log used for generating this model was measured using the so-called *Improved Continuous Semantic Fitness* (De Medeiros, 2006; range:  $-\infty$  to 1.0). This fitness measure indicates the number of correct parsed event sequences, whereas a punishment for allowed extra behaviour in the model is subtracted from this number. The idea of this measure is to favour a process model that allows for less extra behaviour if several models can correctly parse the same number of event sequences. Both process models show a substantial fitness value: the experimental group model = 0.53, and the control group model = 0.62.

#### 4.3.1 *Process model of the experimental group*

For the experimental group, a common pattern — that is, a path of transitions with high certainty — is ANALYZE → PROCESS → SEARCH → REPEAT → EVAL → READ → MONITOR → ANALYZE. Moreover, the process model comprises a number of loops with high certainty between two activities. Participants circle between ANALYZE and PROCESS, EVAL and REPEAT, SEARCH and PROCESS, EVAL and READING, and SEARCH and REPEAT. Apparently, these loops always occur between metacognitive and cognitive learning activities but never between two cognitive or two metacognitive events. Furthermore, it is interesting that EVAL is connected with several other learning events, meaning it takes an important position in the structure of the process, although this event class has a relatively low frequency. MONITOR only shows a weak connection in the process model. This event class follows READ and is followed by ANALYZE. Finally, the model shows self-loops for all event classes, indicating that an activity can be performed multiple times in a row.

#### 4.3.2 *Process model of the control group*

The model of the control group shows the most common path of transitions for SEARCH → PROCESS → EVAL → REPEAT → READ → MONITOR → SEARCH. In contrast to the model of the students in the experimental group, ANALYZE is only weakly connected with SEARCH, and therefore, it is quite isolated. Similar to the experimental group, the low-frequency event class EVAL is also connected with several other learning activities. MONITOR is only weakly connected, whereas this event class follows READ, as in the model of the experimental group, but is followed by SEARCH instead of ANALYZE. In comparison with the experimental group, this process model shows fewer loops with high certainty between two activities (only between SEARCH and PROCESS, SEARCH and REPEAT, and EVAL and REPEAT), but again

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

these loops only occur between metacognitive and cognitive events. Again, all event classes show self-loops.

Overall, the process models of the experimental and control groups especially differ in two points. First, ANALYZE (including the activities Orientation, Planning, and Goal specification) is hardly connected in the model of the control group, but this event class is well embedded in the process of the experimental group. Second, students in the experimental group show more loops between metacognitive and cognitive events, which can be interpreted as “regulation circles.” For example, they circle with high certainty between ANALYZE and PROCESS and between EVAL and READ. Despite these differences, both models have in common that EVAL takes an important position in the described process. The frequency analysis could not reveal the importance of this event class, even showing that EVAL is one of the least-frequent categories. Here, the analysis of the sequential order provides additional information. Moreover, in both models, MONITOR, the metacognitive category with the highest frequency, is hardly connected with other learning activities. Based on recent SRL models, we argue that MONITOR does not have a clear position but can follow each learning activity (e.g.,  $A \rightarrow \text{MONITORING} \rightarrow B \rightarrow \text{MONITORING} \rightarrow C \rightarrow \text{MONITORING}$ ). The HeuristicMiner algorithm could have failed to position this activity in the process model because its modelling notation does not allow for so-called *duplicate tasks* (i.e., an activity that has more than one label in the process model).

## 5 DISCUSSION AND IMPLICATIONS FOR FUTURE RESEARCH

In this study, we analyzed think-aloud data from an experimental study to investigate the effects of metacognitive prompts during learning on SRL processes. In addition to an analysis of frequencies of learning events, we focused on exploring the sequential structure of regulation activities using PM techniques.

As expected, the analysis of coded think-aloud data provides deeper insights into the effects of metacognitive prompts on students’ regulatory processes during hypermedia learning. The findings of a frequency analysis indicate differences in the number of metacognitive utterances between students in the experimental group, who were prompted by metacognitive prompts, and those in the control group, who learned without prompts. Participants supported by metacognitive prompts articulated significantly more metacognitive activities and achieved better transfer performance. In addition, a mediation analysis revealed that prompting increased the number of metacognitive activities, especially Monitoring, which, in turn, increased the transfer performance. Both results are in line with findings of research on metacognitive prompting (e.g., Azevedo et al., 2004; Bannert, 2009) and the assumed effect mechanism of this type of metacognitive support (e.g., Bannert & Mengelkamp, 2013).

A microanalysis of the relative arrangement of learning activities was conducted by means of PM techniques to discover specific sequential patterns in the learning process of the experimental group versus the control group. This process analysis provided additional information on the effects of metacognitive prompts that could not be revealed by a simple analysis of frequencies of occurring

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

learning events. A comparison of the process models of students in the experimental and the control group showed two striking differences. First, activities of orientation, planning, and goal specification (aggregated as ANALYZE) are much better integrated in the process model of the experimental group, whereas this event class was quite isolated in the process model of the control group. Second, more loops between cognitive and metacognitive learning activities were identified in the process model of the experimental group, indicating that more regulation steps occurred. In conclusion, these differences indicate that the use of metacognitive prompts resulted in a better integration of ANALYZE events and a higher number of regulation loops. SRL models (e.g., Winne & Hadwin, 2008; Zimmerman, 2008) emphasize both the importance of orientation phases and an active regulation for successful learning. Following this, the fostered process patterns in the model of the experimental group are in line with current theoretical assumptions. We conclude that these process patterns could be successfully scaffolded through the application of metacognitive prompts. However, the evaluation of learning progress (EVAL) is similarly integrated in both process models, meaning no different effect on this event category could be detected. The findings of the process patterns could be used to optimize our metacognitive prompts further. For example, the design of prompts could be optimized by aiming at scaffolding the sequential deployment of evaluating activities in more detail. SRL models suggest that evaluation activities are followed by an update of the orientation phase. This transition was not represented in the process model of the experimental group. Here, an optimization process could be used.

With respect to the metacognitive support used in this study — that is, an introduction about what metacognitive prompts are, why they are important, and how to use them in combination with metacognitive prompts during learning — it is necessary to discuss which components of support have contributed to the findings. Based on our experience with metacognitive prompts and research on metacognitive prompting, at least a brief training or an introduction to the concept of metacognitive prompts is necessary in advance to guarantee an adequate application of prompts during learning (e.g., Bannert & Mengelkamp, 2013; Bannert & Reimann, 2012). Therefore, it is challenging to determine the individual effect of both the introduction and prompting components. To our knowledge, there is no empirical study that systematically compares the impact of training of prompt use, metacognitive prompting, and their combination. Consequently, this research question should be addressed in future work.

As an inductive approach, the validity of PM depends on the representativity and quality of the data stored in the event log (Reimann, Frerejean, & Thompson, 2009). It is possible that SRL processes, for example, those obtained by think-aloud protocols or log files, comprise a high variety of regulatory behaviour (Hadwin, Nesbit, Jamieson-Noel, Code, & Winne, 2007; Winne, 2014). Therefore, we applied a technique for trace clustering in combination with a PM algorithm to check whether a single process model for the whole event log is appropriate. The applied trace clustering did not split the cases into subsets of participants. However, new approaches of trace clustering are currently rising in the PM domain (e.g., de Weerd et al., 2013). These approaches could possibly improve the detection of

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

homogenous subsamples, which in turn could enhance the quality of the mined process models. There are approaches to clustering students according to their interactions and activities in computer-based learning environments based on a set of variables (e.g., Biswas, Jeong, Kinnebrew, Sulcer, & Roscoe, 2010; Bouchet, Harley, Trevors, & Azevedo, 2013). However, these approaches do not explicitly include an event-centred perspective and a timing aspect, but are based on frequencies of interactions with the learning environment. Furthermore, a subset of participants can also be selected for process analysis based on learner characteristics (e.g., high vs. low prior knowledge) or learning outcomes (e.g., high vs. low achieving students; see Bannert et al., 2014 for an example).

Regarding further limitations of our analysis, it has to be noted that the resulting process models are dependent on the learning setting (learning environment, learning material, and instructions on the learning task). In addition, they are descriptive models. Moreover, findings depend on the underlying coding scheme and its level of granularity. In general, more research on PM techniques in the field of SRL and metacognition is needed; for example, for deriving guidelines for parameter settings aiming to improve the quality of a mined process model. Therefore, we encourage other researchers to use PM techniques to analyze their process data with respect to the sequential and temporal characteristics of learning events. In addition, a comparison of different methods for sequential and temporal analyses on the same data would be beneficial for discovering the advantages and disadvantages of recent process analysis methods. For example, a comparison could be made of different approaches presented in a special issue on the sequential and temporal characteristics of self- and socially regulated learning (Molenaar & Järvelä, 2014).

The resulting process models of our analysis represent a description of the underlying learning processes in our sample of students. In future studies, the validity of the discovered process patterns should be investigated by checking the conformance of these models to new data sets. For this purpose, the mined process models of the HeuristicsMiner algorithm can be converted into petri nets, and then methods for conformance checking can be applied within the ProM framework (Rozinat & van der Aalst, 2008). In this way, the conformance — that is, the differences between a discovered process model and a new event log — can be determined. Another possible scenario for the application of conformance checking would be the derivation of a system of event sequences on the micro-level based on the theoretical assumptions of SRL models. An illustration of this approach is presented in Bannert et al. (2014). However, more micro-level theories would be needed for this approach. At the moment, only the COPES model (Winne & Hadwin, 2008) provides a detailed level of granularity regarding information processing, but even this model is far from the level of elaboration needed to correspond directly to the granularity of our event data.

Finally, an advantage of PM techniques is the representation of sequential characteristics as visual process models. Primarily, this helps the researcher to grasp easily the course of learning activities and regulatory patterns. However, this type of visual representation can also be used to give process feedback to the students and, thereby, be a resource for learners as well (Reimann et al., 2009). Winne



(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

(2014) recommends supporting students with information on their past learning processes through displays of traces they can interpret. Following this direction in future research on SRL, process models generated by PM techniques could make a substantial contribution in providing feedback to learners.

## ACKNOWLEDGMENTS

This research was supported by funds from the German Research Foundation (DFG: BA 2044/7-1). We thank Ferdinand Guggenberger, Valentina Haspert, Michaela Hofbauer and Lisa Müsse, the student research assistants of this project, for their valuable help for coding the think-aloud data.

## REFERENCES

- Azevedo, R. (2005). Using hypermedia as a metacognitive tool for enhancing student learning? The role of self-regulated learning. *Educational Psychologist*, 40(4), 199–209. doi:10.1207/s15326985ep4004\_2
- Azevedo, R. (2009). Theoretical, conceptual, methodological, and instructional issues in research on metacognition and self-regulated learning: A discussion. *Metacognition and Learning*, 4(1), 87–95. doi:10.1007/s11409-009-9035-7
- Azevedo, R., Cromley, J. G., Moos, D. C., Greene, J. A., & Winters, F. I. (2011). Adaptive content and process scaffolding: A key to facilitating students' self-regulated learning with hypermedia. *Psychological Testing and Assessment Modeling*, 53(1), 106–140.
- Azevedo, R., Guthrie, J., & Seibert, D. (2004). The role of self-regulated learning in fostering students' conceptual understanding of complex systems with hypermedia. *Journal of Educational Computing Research*, 30(1–2), 87–111. doi:10.2190/DVWX-GM1T-6THQ-5WC7
- Azevedo, R., Trevors, G., Duffy, M., Feyzi-Behnagh, M., Bouchet, F., & Landis, R. (2013). Using trace data to examine the complex roles of cognitive, metacognitive, and emotional self-regulatory processes during learning with multi-agent systems. In R. Azevedo & V. Aleven (Eds.), *International Handbook of Metacognition and Learning Technologies* (pp. 427–449). New York, NY: Springer. doi:10.1007/978-1-4419-5546-3\_28
- Bannert, M. (2006). Effects of reflection prompts when learning with hypermedia. *Journal of Educational Computing Research*, 35(4), 359–375. doi:10.2190/94V6-R58H-3367-G388
- Bannert, M. (2007). *Metakognition beim Lernen mit Hypermedia. Erfassung, Beschreibung und Vermittlung wirksamer metakognitiver Lernstrategien und Regulationsaktivitäten [Metacognition and hypermedia learning]*. Münster: Waxmann.
- Bannert, M. (2009). Promoting self-regulated learning through prompts: A discussion. *Zeitschrift für Pädagogische Psychologie / German Journal of Educational Psychology*, 23(2), 139–145. doi:10.1024/1010-0652.23.2.139
- Bannert, M., & Mengelkamp, C. (2013). Scaffolding hypermedia learning through metacognitive prompts. In R. Azevedo & V. Aleven (Eds.), *International handbook of metacognition and learning technologies* (pp. 171–186). New York, NY: Springer. doi:10.1007/978-1-4419-5546-3\_12

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

- Bannert, M., & Reimann, P. (2012). Supporting self-regulated hypermedia learning through prompts. *Instructional Science*, 40(1), 193–211. doi:10.1007/s11251-011-9167-4
- Bannert, M., Reimann, P., & Sonnenberg, C. (2014). Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and Learning*, 9(2), 161–185. doi:10.1007/s11409-013-9107-6
- Bannert, M., Sonnenberg, C., Mengelkamp, C., & Pieger, E. (2015). Short- and long-term effects of students' self-directed metacognitive prompts on navigation behavior and learning performance. *Computers in Human Behavior*, 52, 293–306. doi:10.1016/j.chb.2015.05.038
- Biswas, G., Jeong, H., Kinnebrew, J. S., Sulcer, B., & Roscoe, R. (2010). Measuring self-regulated learning skills through social interactions in a teachable agent environment. *Research and Practice in Technology Enhanced Learning*, 5(2), 123–152. doi:10.1142/S1793206810000839
- Bloom, B. S. (1956). *Taxonomy of educational objectives: The classification of educational goals*. New York, NY: D. McKay.
- Boekaerts, M. (1997). Self-regulated learning: A new concept embraced by researchers, policy makers, educators, teachers, and students. *Learning and Instruction*, 7(2), 161–186. doi:10.1016/S0959-4752(96)00015-1
- Bouchet, F., Harley, J. M., Trevors, G. J., & Azevedo, R. (2013). Clustering and profiling students according to their interactions with an intelligent tutoring system fostering self-regulated learning. *Journal of Educational Data Mining*, 5(1), 104–146.
- Chi, M. T. H. (1997). Quantifying qualitative analyses of verbal data: A practical guide. *Journal of the Learning Sciences*, 6(3), 271–315. doi:10.1207/s15327809jls0603\_1
- De Medeiros, A. K. A. (2006). *Genetic Process Mining*. Eindhoven University of Technology, Eindhoven.
- De Medeiros, A. K. A., Guzzo, A., Greco, G., van der Aalst, W. M. P., Weijters, A. J. M. M., van Dongen, B. F., & Saccà, D. (2008). Process mining based on clustering: A quest for precision. *Lecture Notes in Computer Science*, 4928, 17–29. doi:10.1007/978-3-540-78238-4\_4
- De Weerd, J., de Backer, M., Vanthienen, J., & Baesens, B. (2012). A multi-dimensional quality assessment of state-of-the-art process discovery algorithms using real-life event logs. *Information Systems*, 37(7), 654–676. doi:10.1016/j.is.2012.02.004
- De Weerd, J., vanden Broucke, S., Vanthienen, J., & Baesens, B. (2013). Active trace clustering for improved process discovery. *IEEE Transactions on Knowledge and Data Engineering*, 25(10), 2708–2720. doi:10.1109/TKDE.2013.64
- Ericsson, K. A., & Simon, H. A. (1993). *Protocol analysis: Verbal reports as data*. Cambridge, MA: MIT Press.
- Fluxicon Disco (Version 1.7.2) [Computer Software]. (2014). Retrieved from <http://fluxicon.com/disco/>
- Ge, X. (2013). Designing learning technologies to support self-regulation during ill-structured problem-solving processes. In R. Azevedo & V. Aleven (Eds.), *International handbook of metacognition and learning technologies* (pp. 213–228). New York, NY: Springer. doi:10.1007/978-1-4419-5546-3\_15

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

- Greco, G., Guzzo, A., Pontieri, L., & Saccà, D. (2006). Discovering expressive process models by clustering log traces. *IEEE Transactions on Knowledge and Data Engineering*, 18(8), 1010–1027. doi:10.1109/TKDE.2006.123
- Greene, J. A., Dellinger, K. R., Tüysüzoglu, B. B., & Costa, L.-J. (2013). A two-tiered approach to analyzing self-regulated learning data to inform the design of hypermedia learning environments. In R. Azevedo & V. Alevan (Eds.), *International handbook of metacognition and learning technologies* (pp. 117–128). New York, NY: Springer. doi:10.1007/978-1-4419-5546-3\_8
- Günther, C. W., & van der Aalst, W. M. P. (2007). Fuzzy mining: Adaptive process simplification based on multi-perspective metrics. *Business Process Management: Lecture Notes in Computer Science*, 4714, 328–343. doi:10.1007/978-3-540-75183-0
- Hadwin, A. F., Nesbit, J. C., Jamieson-Noel, D., Code, J., & Winne, P. H. (2007). Examining trace data to explore self-regulated learning. *Metacognition and Learning*, 2(2), 107–124. doi:10.1007/s11409-007-9016-7
- Hayes, A. F. (2013). *Introduction to mediation, moderation, and conditional process analysis: A regression-based approach*. New York, NY: The Guilford Press.
- Hoffman, B., & Spatariu, A. (2011). Metacognitive prompts and mental multiplication: Analyzing strategies with a qualitative lens. *Journal of Interactive Learning Research*, 22(4), 607–635.
- Jeong, H., Gupta, A., Roscoe, R., Wagster, J., Biswas, G., & Schwartz, D. (2008). Using hidden Markov models to characterize student behaviors in learning-by-teaching environments. *Lecture Notes in Computer Science*, 5091, 614–625. doi:10.1007/978-3-540-69132-7\_64
- Johnson, A. M., Azevedo, R., & D’Mello, S. K. (2011). The temporal and dynamic nature of self-regulatory processes during independent and externally assisted hypermedia learning. *Cognition and Instruction*, 29(4), 471–504. doi:10.1080/07370008.2011.610244
- Kapur, M. (2011). Temporality matters: Advancing a method for analyzing problem-solving processes in a computer-supported collaborative environment. *International Journal of Computer-Supported Collaborative Learning*, 6(1), 39–56. doi:10.1007/s11412-011-9109-9
- Li, C., & Biswas, G. (2002). A Bayesian approach for structural learning with hidden Markov models. *Scientific Programming*, 10(3), 201–219.
- Lin, X. (2001). Designing metacognitive activities. *Educational Technology Research and Development*, 49(2), 23–40. doi:10.1007/BF02504926
- Lin, X., Hmelo, C., Kinzer, C. K., & Secules, T. J. (1999). Designing technology to support reflection. *Educational Technology Research and Development*, 47(3), 43–62. doi:10.1007/BF02299633
- Lin, X., & Lehman, J. D. (1999). Supporting learning of variable control in a computer-based biology environment: Effects of prompting college students to reflect on their own thinking. *Journal of Research in Science Teaching*, 36(7), 837–858. doi:10.1002/(SICI)1098-2736(199909)36:7<837::AID-TEA6>3.3.CO;2-L
- Martin, T., & Sherin, B. (2013). Learning analytics and computational techniques for detecting and evaluating patterns in learning: An introduction to the special issue. *Journal of the Learning Sciences*, 22(4), 511–520. doi:10.1080/10508406.2013.840466

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

- Molenaar, I., & Chiu, M. M. (2014). Dissecting sequences of regulation and cognition: Statistical discourse analysis of primary school children's collaborative learning. *Metacognition and Learning*, 9(2), 137–160. doi:10.1007/s11409-013-9105-8
- Molenaar, I., & Järvelä, S. (2014). Sequential and temporal characteristics of self and socially regulated learning. *Metacognition and Learning*, 9(2), 75–85. doi:10.1007/s11409-014-9114-2
- Moos, D. C., & Azevedo, R. (2009). Self-efficacy and prior domain knowledge: To what extent does monitoring mediate their relationship with hypermedia learning? *Metacognition and Learning*, 4(3), 197–216. doi:10.1007/s11409-009-9045-5
- Preacher, K. J., & Kelley, K. (2011). Effect size measures for mediation models: Quantitative strategies for communicating indirect effects. *Psychological Methods*, 16(2), 93–115. doi:10.1037/a0022658
- ProM (Version 5.2) [Computer Software]. (2008). Retrieved from <http://www.processmining.org>
- Reimann, P. (2009). Time is precious: Variable- and event-centred approaches to process analysis in CSCL research. *International Journal of Computer-Supported Collaborative Learning*, 4(3), 239–257. doi:10.1007/s11412-009-9070-z
- Reimann, P., Frerejean, J., & Thompson, K. (2009). Using process mining to identify models of group decision making processes in chat data. In C. O'Malley, D. Suthers, P. Reimann, & A. Dimitracopoulou (Eds.), *Computer-Supported Collaborative Learning Practices: CSCL2009 Conference Proceedings* (pp. 98–107). Rhodes, Greece: International Society for the Learning Sciences.
- Reimann, P., & Yacef, K. (2013). Using process mining for understanding learning. In R. Luckin, P. Puntambekar, P. Goodyear, B. Grabowski, J. Underwood, & N. Winters (Eds.), *Handbook of design in educational technology* (pp. 472–481). New York, NY: Routledge.
- Reisig, W. (1985). *Petri Nets: An introduction*. Berlin: Springer.
- Richter, T., & van Holt, N. (2005). ELVES: Ein computergestütztes Diagnostikum zur Erfassung der Effizienz von Teilprozessen des Leseverstehens [ELVES: Computer-based assessment of the efficiency of processes involved in reading comprehension]. *Diagnostica*, 51(4), 169–182. doi:10.1026/0012-1924.51.4.169
- Romero, C., Ventura, S., Pechenizkiy, M., & Baker, R. (2010). *Handbook of educational data mining*. Boca Raton, FL: Chapman & Hall/CRC.
- Rozinat, A. (2010, October 18). ProM tips: Which mining algorithm should you use? [Web log post]. Retrieved from <http://fluxicon.com/blog/2010/10/prom-tips-mining-algorithm/>
- Rozinat, A., & van der Aalst, W. M. P. (2008). Conformance checking of processes based on monitoring real behavior. *Information Systems*, 33(1), 64–95. doi:10.1016/j.is.2007.07.001
- Schnotz, W. (1998). Strategy-specific information access in knowledge acquisition from hypertext. In L. B. Resnick, R. Säljö, C. Pontecorvo, & B. Burge (Eds.), *Discourse, tools, and reasoning: Essays on situated cognition* (pp. 336–358). Berlin: Springer.
- Schoor, C., & Bannert, M. (2012). Exploring regulatory processes during a computer-supported collaborative learning task using process mining. *Computers in Human Behavior*, 28(4), 1321–1331. doi:10.1016/j.chb.2012.02.016

(2015). Discovering the Effects of Metacognitive Prompts on the Sequential Structure of SRL-Processes Using Process Mining Techniques. *Journal of Learning Analytics*, 2(1), 72–100.

- Trčka, N., Pechenizkiy, M., & van der Aalst, W. M. P. (2010). Process mining from educational data. In C. Romero, S. Ventura, M. Pechenizkiy, & R. Baker (Eds.), *Handbook of educational data mining* (pp. 123–142). Boca Raton, FL: Chapman & Hall/CRC.
- Van der Aalst, W. M. P. (2011). *Process mining: Discovery, conformance and enhancement of business processes*. Berlin/Heidelberg: Springer. doi:10.1007/978-3-642-19345-3
- Van der Aalst, W., Weijters, T., & Maruster, L. (2004). Workflow mining: Discovering process models from event logs. *IEEE Transactions on Knowledge and Data Engineering*, 16(9), 1128–1142. doi:10.1109/TKDE.2004.47
- Veenman, M. (1993). *Metacognitive ability and metacognitive skill: Determinants of discovery learning in computerized learning environments*. Amsterdam: University of Amsterdam.
- Veenman, M. V. J. (2007). The assessment and instruction of self-regulation in computer-based environments: A discussion. *Metacognition and Learning*, 2(2–3), 177–183. doi:10.1007/s11409-007-9017-6
- Veenman, M. V. J., van Hout-Wolters, B. H. A. M., & Afflerbach, P. (2006). Metacognition and learning: Conceptual and methodological considerations. *Metacognition and Learning*, 1(1), 3–14. doi:10.1007/s11409-006-6893-0
- Weijters, A. J. M. M., van der Aalst, W. M. P., & de Medeiros, A. K. A. (2006). Process mining with the HeuristicsMiner algorithm. *BETA Working Paper Series 166*. Eindhoven, Netherlands: Eindhoven University of Technology.
- Winne, P. H. (2014). Issues in researching self-regulated learning as patterns of events. *Metacognition and Learning*, 9(2), 229–237. doi:10.1007/s11409-014-9113-3
- Winne, P. H., & Hadwin, A. F. (2008). The weave of motivation and self-regulated learning. In D. Schunk & B. J. Zimmerman (Eds.), *Motivation and self-regulated learning: Theory, research, and applications* (pp. 297–314). New York, NY: Taylor & Francis.
- Winne, P. H., & Hadwin, A. F. (2013). nStudy: Tracing and supporting self-regulated learning in the internet. In R. Azevedo & V. Aleven (Eds.), *International handbook of metacognition and learning technologies* (pp. 293–308). New York, NY: Springer. doi:10.1007/978-1-4419-5546-3\_20
- Winne, P. H., & Nesbit, J. C. (2009). Supporting self-regulated learning with cognitive tools. In D. J. Hacker, J. Dunlosky, & A. C. Graesser (Eds.), *Handbook of metacognition in education* (pp. 259–277). New York, NY: Routledge.
- Winne, P. H., & Perry, N. E. (2000). Measuring self-regulated learning. In M. Boekaerts, P. Pintrich, & M. Zeidner (Eds.), *Handbook of self-regulation* (pp. 531–566). San Diego, CA: Academic Press.
- Zimmerman, B. J. (2000). Attaining self-regulation: A social cognitive perspective. In M. Boekaerts, P. Pintrich, & M. Zeidner (Eds.), *Handbook of self-regulation* (pp. 13–39). San Diego, CA: Academic Press.
- Zimmerman, B. J. (2008). Investigating self-regulation and motivation: Historical background, methodological developments, and future prospects. *American Educational Research Journal*, 45(1), 166–183. doi:10.3102/000283120731290