Using Large Language Models to Detect Self-Regulated Learning in Think-Aloud Protocols

Jiayi Zhang[1], Conrad Borchers[2], Vincent Aleven[2], Ryan S. Baker[1]
[1]University of Pennsylvania
[2]Carnegie Mellon University
joycez@upenn.edu

ABSTRACT
Think-aloud protocols are a common method to study self-regulated learning (SRL) during learning by problem-solving. Previous studies have manually transcribed and coded students' verbalizations, labeling the presence or absence of SRL strategies and then examined these SRL codes in relation to learning. However, the coding process is difficult to scale, as it is time-consuming and laborious. This aspect potentially limits the ability to measure SRL comprehensively on a larger scale. Recent advancements in language models offer the potential to infer SRL from automated think-aloud transcriptions, which could enhance the efficiency of SRL measurement, complementing log data-based approaches to studying SRL. Therefore, this study explores the possibility of leveraging large language models (LLMs) and machine learning to automatically detect SRL in machine-transcribed student think-aloud transcripts. Specifically, we experimented with two LLMs (Universal Sentence Encoders and OpenAI's text-embedding-3-small) to predict four SRL categories (processing information, planning, enacting, and realizing errors) in students' verbalizations, collected from three intelligent tutoring systems, covering stoichiometry chemistry and formal logic. We found that these models are reliable at predicting the SRL categories, with AUC scores ranging from 0.696 to 0.915. Models that use embeddings from the text-embedding-3-short model performed significantly better at predicting SRL, including transfer from open-ended to highly scaffolded ITS systems. However, we note limitations in transferring models from the chemistry to logic domain, potentially due to the differences in domain-specific vocabulary. We discuss the practical implications of these models, highlighting the opportunity to analyze think-aloud transcripts at scale to facilitate future SRL research.

Keywords
Self-regulated learning, think-aloud protocols, large language models, NLP, intelligent tutoring system

1. INTRODUCTION
Successful use of self-regulated learning (SRL) has frequently been found to be positively associated with learning and learning outcomes [14, 23, 46, 68]. Students who are skilled in SRL are able to effectively set goals [37], search for information [67] , and direct their attention and cognitive resources to align their efforts with their objectives [29, 67].

Given the importance of SRL in the learning process, prior studies have utilized **behavioral log data** to measure and facilitate students' use of SRL within intelligent tutoring systems (ITSs). These adaptive and personalized software systems are designed to offer step-by-step guidance throughout problem-solving tasks [2]. By analyzing the patterns of behaviors from students interacting with an ITS, researchers can draw inferences, identifying which SRL strategies the students are using, how they are used, and in what order [3, 51, 52, 58, 59]. With this approach, previous studies have used behavioral log data to examine a range of SRL behaviors, including help-seeking behaviors [1], gaming the system (an ineffective use of SRL; [8]), setting goals, making plans [4, 10], tracking progress [10], and engaging in various cognitive operations, such as assembling and monitoring, during problem-solving [31, 44, 63]. Automated detectors have been developed to measure these SRL behaviors in an immediate fashion, offering assessments that identify both the SRL behaviors students are employing and those they may be lacking.

In addition to log data, **think-aloud protocols (TAPs)** are another approach that has been frequently used in previous studies for measuring SRL *in situ* [9, 25, 26, 37]. During think-aloud activities, students are asked to verbalize their thinking and cognitive processes as they interact with an ITS while solving a problem. Utterances collected from think-aloud activities are then transcribed and segmented into clips. To assess students' use of self-regulation, researchers manually code students' verbalizations in each clip, labeling the presence or absence of SRL strategies [25]. Using this approach, previous studies have examined how the use of SRL in terms of presence, frequency, and the sequential and temporal order relate to the overall learning outcomes (e.g., [9, 37, 42]) and to the moment-by-moment performance when solving a multi-step problem [12].

However, the coding process in TAPs is difficult to scale, as it is time-consuming and laborious. This aspect potentially limits the ability to measure SRL comprehensively on a larger scale in SRL research using the TAP approach. Having the capability to measure SRL at scale in think-aloud data presents an opportunity to explore a wide range of SRL behaviors across different contexts on a larger scale.

Being able to measure SRL in think-aloud at scale is particularly relevant given the two approaches—behavioral log data and think-aloud protocols—appear to complement each other rather than being substitutive, as noted in [19]. In their study, they show that SRL behaviors, such as orientation and elaboration, were more likely to be detected in behavioral log data than in think-aloud protocols. This could potentially be due to the differences in students' ability to articulate their thoughts in the form of useful data, particularly when students are experiencing high cognitive demands during a think-aloud activity, in which they are learning while verbalizing their thoughts [19]. In contrast, SRL behaviors, such as planning and monitoring, were found to be more easily

identified through think-aloud protocols. These behaviors involve the engagement of metacognitive processes that are often challenging to detect solely through log data [17, 43]. This finding further highlights the unique and significant role of TAPs in SRL measurement.

Given the importance of using TAPs to measure and understand SRL and the limitations with analyzing TAPs at scale, the current study explores the possibilities of leveraging large language models (LLMs) and machine learning to automatically detect SRL behaviors in machine-generated student think-aloud transcripts. Specifically, we collected students' think aloud data from three intelligent tutoring systems covering stoichiometry chemistry and formal logic. The audio was transcribed using Whisper, a state-of-the-art speech-to-text software. After analyzing the transcripts, we operationalized four SRL categories—*Processing Information, Planning, Enacting, and Realizing Errors*—grounded in Winne and Hadwin's four-stage model [60], representing key behaviors in each step. We then conducted a round of coding, labeling the presence or absence of the four SRL categories using a coding scheme. Two sentence embedding models (Universal Sentence Encoders v5 [13] and Open AI's text-embedding-3-small [45]) were applied respectively to vectorize the text. Using the outputs from the embedding models as features, we trained machine learning models that predict the presence or absence of the four SRL categories.

While theoretical models of SRL are agnostic to ITS systems and their domain of instruction, the generalizability of machine-learned models of SRL trained on think-aloud transcriptions is an open research question. Specifically, models trained on sentence embeddings might pick up on semantics specific to a domain (e.g., finding the reactant in a chemistry problem compared to planning to learn a transformation in formal logic). However, it is desirable for a language-based SRL model to be domain-independent, not only because theoretical SRL models are domain-agnostic but also because a domain-general SRL model could be adapted and plucked into novel learning contexts in a low-cost manner. In other words, if a general SRL model based on think-aloud data is highly generalizable, no new training data needs to be labeled for new ITS environments and domain contexts. While that is desirable from an economic standpoint (as labeling data and training models is costly) past work in educational data mining and learning analytics has also described domain transfer as a grand challenge of the field [7]. Therefore, we systematically evaluated our trained model across the domains of chemistry and formal logic. Similarly, domain transfer might depend on the similarity of tutoring system interfaces and architectures [49]. To investigate this possibility, we evaluated the robustness of our model across open-ended, formula-based ITSs compared to a highly structured ITS with fraction-based input.

The present study's contributions are twofold. We hope to 1) demonstrate the possibility of automating SRL measurement in think-aloud data, and 2) examine the transferability of these models across subject areas and platform designs Having the capability to measure SRL at scale in think-aloud data presents an opportunity to explore a wide range of SRL behaviors across different contexts on a larger scale.

2. BACKGROUND

2.1 Self-regulated Learning

Self-regulation, a critical component in learning, is where learners take active control of their learning by monitoring and regulating their attention and effort in pursuit of goals [69]. During this process, learners may set goals, monitor progress, and adjust strategies when goals are not met. A range of cognitive, metacognitive, affective, behavioral, and motivational processes are involved in SRL. Engaging in these processes effectively enable learners to become more independent and effective in their learning [67]. In general, students who effectively self-regulate their learning tend to perform better than those who do not [68] and are more likely to have deep conceptual understanding on the topic [5, 24, 36].

In the last three decades, several theoretical models have been proposed from different perspectives to depict the process of SRL [47]. For example, based on socio-cognitive theories, Zimmerman [67] describes the process of SRL as three cyclical phases (i.e., forethought, performance, and self-reflection), in which learners analyze a task, execute the task, and assess and evaluate the performance respectively. Grounded in information processing theory, Winne and Hadwin [60] characterize the process of SRL as four interdependent and recursive stages, in which learners: 1) define the task, 2) set goals and form plans, 3) enact the plans, and 4) reflect and adapt strategies when goals are not met. A range of SRL behaviors may be involved in each stage of the cycle.

Despite the differences in theoretical groundings and focuses, most of the models describe SRL as a cyclical process consisting of phases where learners understand tasks, make plans, enact the plans, and reflect and adapt [37, 57, 58]. These theoretical models are frequently adopted in recent SRL research as foundations that guide the conceptualization and operationalization of SRL in SRL measurement [57, 66]. Recent work in educational data mining and learning analytics has provided empirical support for cyclical models of SRL by relating cyclical SRL stages to learner performance data [9, 12, 27, 29].

2.2 Using Think-aloud Protocols to Measure and Understand SRL

Grounding the analysis and operationalization in these SRL theories, previous studies have used think-aloud protocols to measure and understand SRL behaviors and processes. In think-aloud activities, students are asked to verbalize their thinking and cognitive processes while they solve a problem [25]. Utterances collected from think-aloud allow researchers to measure and examine SRL behaviors that are contextualized in the problem-solving process and are approximately concurrent with their occurrences.

To engage students in think-aloud activities, instructions are often given prior to a task, asking students to verbalize their thinking while working on a task, as if they are speaking to themselves [18]. Once the task begins, researchers or the learning software may use simple prompts such as "please keep talking" to remind participants to continue to talk, when learners stop verbalizing [26]. These instructions and prompts are designed with the goal of inflicting a minimum amount of distraction without altering a student's thinking process.

To accurately capture students' thinking process, Ericsson & Simon [18] provide guidelines on the TAP instructions and prompts. In this, they contend that prompts should primarily focus on asking students to express conscious thoughts using language that directly represents those thoughts (e.g., "my plan is to complete the assignment") or express thoughts in which sensory information is converted into words (e.g., "I see three hyperlinks here"). In contrast, prompts should refrain from asking students to

metacognitively monitor and reflect on their thinking process, as this can potentially influence how students think and perform tasks, altering the order and nature of their cognitive processes [18, 53]. When prompts are carefully designed to avoid engaging students in metacognitive activities, studies have found that thinking aloud neither alters accuracy nor the sequence of operations in most cognitive tasks (excluding insight problems [21]).

Once students complete the learning task, their verbalization collected using audio or video recordings is then transcribed to text. The recordings, once predominantly transcribed by humans, are now increasingly transcribed by automated transcription tools such as Whisper [50], with transcription accuracy described in their technical reports being satisfactory without human supervision.

With the transcriptions, researchers code the SRL processes using a coding scheme (e.g., [9, 29, 37]). As a critical part in TAP, the coding scheme outlines the target SRL behaviors to observe in a transcript and provide an operationalization for each behavior. These schemes are typically derived from SRL theories and then refined and modified based on the existing task, platform, and dataset [26].

For example, to examine students' use of SRL in think-aloud transcript, [9] developed a coding scheme that outlines SRL categories corresponding to the three phases (i.e., forethought, performance, and reflection) in Zimmerman's model [67]. These categories reflect behaviors where students are self-regulating in the forethought phase (e.g., orientation, planning, setting goals), performance phase (e.g., processing), and reflection phase (e.g., evaluating). By comparing the SRL activities between high and low-achieving students, [9] found that high achievers tended to demonstrate more frequent use of SRL, such as planning and monitoring; and they are also more effective and strategic at implementing SRL strategies. Using the same coding scheme, [29] and [37] found that successful learners were more likely to engage in preparatory activities (e.g., orientation and planning) before completing a task. In contrast, preparation and evaluation activities were less frequently used by less successful students.

In addition to studying the effective use of SRL in relation to a student's overall achievement, a recent study examined how the use of SRL inferred from TAPs is correlated to moment-by-moment performance when students are solving a multi-step problem [12]. In specific, they identified four SRL categories based on Winne and Hadwin's four-stage model [60]. By coding SRL categories in students' utterances in between steps, they examined how the use of SRL in terms of presence, frequency, cyclical characteristics, and recency relate to student performance on subsequent steps in multi-step problems. They show that students' actions during process-heavy stages of SRL cycles (e.g., processing information and planning) exhibited lower moment-by-moment correctness than later SRL cycle stages (i.e., enacting) during problem-solving. This more granular examination between students' use of SRL and intermediate success provides a lens through which to examine the effectiveness of SRL behaviors in a fine-grained way. Understanding how SRL behaviors influence the subsequent performance provide further evidence on when interventions could be provided during a problem-solving process.

2.3 Using Natural Language Processing to Scale Up SRL Measurement

Six years ago, the position papers published by McNamara et al. [40] discussed the significant impact of natural language processing (NLP) in understanding and facilitating learning, emphasizing that natural language is fundamental to conveying information. Recent advancements in NLP continue to reveal new opportunities for supporting learning through analytics.

One emerging application using natural language in the domain of education is predicting students' cognitive processes from learner text artifacts with the goal to provide real-time feedback and to measure these constructs at scale. For example, to understand how students engage in SRL and to provide timely scaffolds in math problem-solving, using NLP and machine learning, Zhang et al [63] developed detectors that measure SRL in students' open-ended responses. Specifically, they extracted features that resemble the linguistic characteristics found in students' text-based responses and trained machine learning models that detect SRL constructs, reflecting how students assemble information, form mental representations of a problem, and monitor progress. Similarly, Kovanovic et al [35] developed machine learning models that automatically identify the types of reflection in students' reflective writing. In their work, NLP methods, including n-grams, LIWC [54], and Coh-Metrix [15] were used to extract features from students' reflective writing which were then used to train models and make predictions.

Since then, more advanced models have been developed to process human language. Large language models (LLMs) and sentence embeddings as the most recent breakthrough in NLP have advanced the state of the art in language models. These models, based on deep learning architectures such as transformer neural networks, are trained on massive amounts of text data to understand and generate human language in a contextually coherent and meaningful manner [56]. Sentence encoders play a crucial role within LLMs by transforming sentences into embeddings within a high-dimensional vector space. This process is accomplished by leveraging the learned relationships between words from previously encountered sentences. Specifically, these sentence encoders convert text to N-dimensional embeddings by considering each word within a sentence and its context with surrounding words. (e.g., N = 1536 for Open AI's text-embedding-3-short model, N = 768 for BERT-base, N = 512 for Universal Sentence Encoders v5). Consequently, these contextually rich embeddings, expressed in numerical format, capture the semantic meaning and contextual information of the text.

Given these advancements, particularly with LLMs showing promising performance in text comprehension, studies have explored integrating them into detectors to scale up measurement, predicting cognitive constructs in textual artifacts. This includes detecting attributes and relatedness of peer feedback [16, 64], as well as detecting gaming the system (a failure to engage in SRL) in open-ended responses [65].

However, the transferability of detectors has commonly been mentioned as a limitation in EDM and related fields in previous work [7]. The models developed in these papers are mainly designed and evaluated within one platform ([48] represents one of the few exceptions). Being able to evaluate the performance of a detector across systems will allow us to understand the limitations of these models, as well as to investigate how the language/communicative expression may differ when students' working in different subjects and systems, albeit capturing the same cognitive attributes.

3. METHODS

To develop models that automatically detect students' use of SRL in think-aloud data, we collected students' think-aloud transcripts while working within three intelligent tutoring systems (ITSs). In this section, we first describe the three ITSs, summarizing the content and design of each system, and comparing how they differ from each other (see section 3.1). We then describe the study sample and study procedure, highlighting the environment in which think-aloud was collected (see section 3.2). After analyzing the think-aloud transcription, we operationalized four SRL categories grounded in the Winne and Hadwin four-stage model [60], and developed a coding scheme to code the utterances accordingly (see section 3.3). Two embedding models were then applied respectively to vectorize the text. Using the outputs from the embedding models as features, we trained machine learning models that predict the presence or absence of the four SRL categories (see section 3.4). Finally, we examined how well these models transfer across subject areas and platforms (see section 3.5).

3.1 Study Materials

The present study's sample features student interactions from three intelligent tutoring systems, covering the domains of stoichiometry chemistry and formal logic. All tutoring systems provide students with step-level tutoring, including correctness feedback and as-needed instructions in the form of hints, as is common in ITSs. However, the degree of structure varies across these systems, from an open-ended, formula-based ITS to a highly structured ITS with fraction-based input.

Our first system, Stoichiometry Tutor, is an ITS based on example tracing [2]. Stoichiometry Tutor has significantly improved high school students' stoichiometry skills [38, 39]. The most recent version of the Stoichiometry Tutor incorporates insights from previous design experiments, including the use of polite language in hints [38]. The Stoichiometry Tutor utilizes a structured, fraction-based problem-solving method to guide students toward target values (see Figure 1.top).

Our second system, the Open-Response Chemistry Cognitive Assistant (ORCCA) Tutor, is a rule-based ITS for chemistry [33]. As a rule-based ITS, ORCCA matches problem-solving rules with students' strategies, accommodating flexible problem-solving sequences with a formula interface (see Figure 1.middle). Rule-based ITSs allow for more flexibility in problem-solving strategies while at times being unable to match errors to intended strategies, limiting their ability to deliver as-needed feedback.

Our third system, the Logic Tutor, is a rule-based tutoring system for learning propositional logic (see Figure 1.bottom). Students are guided through constructing truth tables, correlating the structure of a formula with its meaning by assigning truth values. Students are tasked with manipulating propositional formulae and transforming them into equivalent expressions using a limited set of logical connectives. Additionally, students learn to apply transformation rules to rewrite or simplify formulas. Students receive hints and error feedback with dynamically generated counterexamples to students' formulae. A cheat sheet on the left reminds them about relevant logical transformations and reason boxes for self-explanation, which pose further scaffolding during problem-solving. The tutoring system has been used in remedial college summer pre-courses for incoming first-year university students.

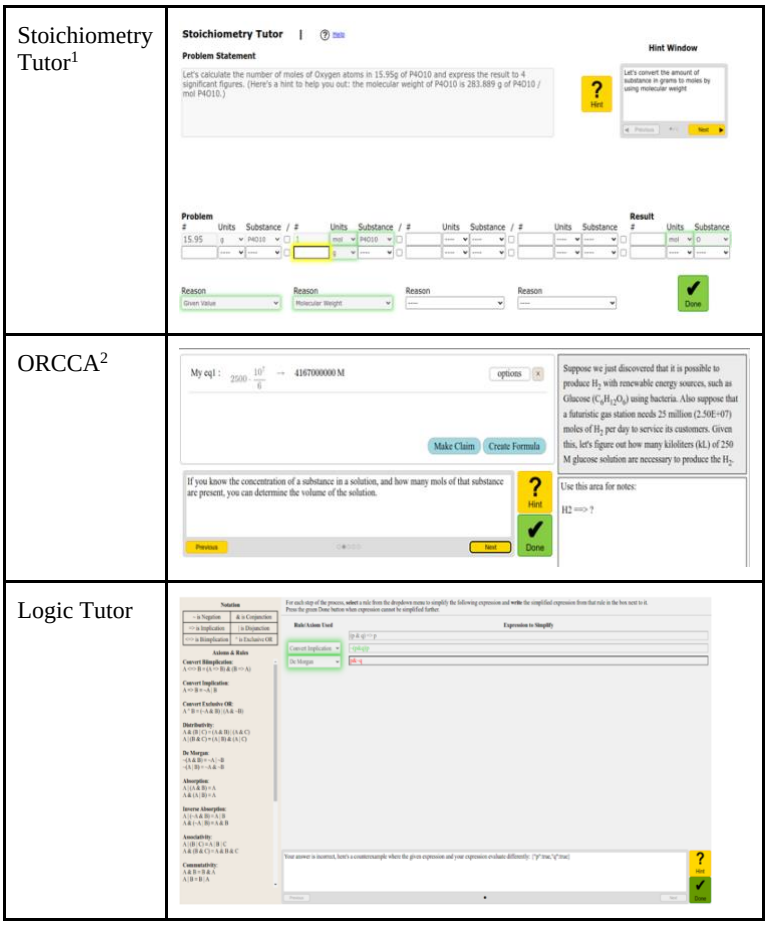

Figure 1. Interface examples of all three ITSs employed in the present study. Stoichiometry Tutor and ORCCA cover the domain of chemistry. Logic Tutor and ORCCA are formula-based ITS compared to Stoichiometry Tutor, which is highly structured, fraction-based ITS.

3.2 Study Sample and Procedure

Fifteen students enrolled in undergraduate (93.3%) and graduate degree (6.7%) programs participated in this study between February and November 2023. Ten students were recruited at a private research university (participating in person). The five other students were recruited from a large public liberal arts university (participating remotely via Zoom). All participants were enrolled in degree programs in the United States. Participants were 40.0% white, 46.6% Asian, and 13.3% multi- or biracial. The students were undergraduate freshmen (21.4%), sophomores (14.3%), juniors (35.7%) and seniors (21.4%) and one graduate student (7.1%). Students were recruited via course-related information channels via instructors and student snowball recruiting. In all cases, recruitment was performed via channels, ensuring that students were still in the process of learning the content domain covered by the tutoring system. All participants received a $15 Amazon gift card for their participation.

All students completed a session between 45-60 minutes. Students were distributed to conditions such that at least five students worked with each of the three ITSs. Six students worked on both tutors for stoichiometry due to finishing their first ITS early. The procedure started with a self-paced questionnaire assessing demographic information, prior academic achievement, and self-rated proficiency in the subject domain (i.e., stoichiometry

[1] https://stoichtutor.cs.cmu.edu/

[2] https://orcca.ctat.cs.cmu.edu/

chemistry or formal logic). Then, students viewed a pre-recorded introductory video about the ITS they would work with and could ask questions about the video. In the case of Logic Tutor, students had the opportunity to read an article on formal logic symbolization and rules and to ask the experimental conductor, who was familiar with formal logic, any questions about symbolization and the content. Students had up to five minutes to skim both articles to develop relevant questions to ask the experimental conductor. Both articles were taken from a remedial first-year undergraduate summer course on formal logic in which the Logic Tutor was previously deployed. The articles ensure that all participants had the necessary prerequisite knowledge and knew the required symbolization for expressing logical formulae to work with the tutoring software. After being acquainted with the tutoring software, students received a brief demonstration and introduction to the think-aloud method and began working on tutor problems at their own pace for up to 20 minutes while thinking aloud. The experimental conductor occasionally reminded them to keep talking when they fell silent for more than 5 s. Think-aloud utterances were recorded with a 2022 MacBook Pro built-in microphone of the computer serving the tutoring software or Zoom microphones of the participating student's laptop.

Problems were content-matched across the Stoichiometry Tutor and ORCCA ITS and included two content units taken from prior studies featuring the Stoichiometry Tutor: (a) moles and gram conversion and (b) stoichiometric conversion. Both content units included a total of four problems. The ordering of problems was counterbalanced by reversing the problem sequence across all four conditions. For Logic Tutor, two problem sets were taken from prior remedial summer courses for first-year undergraduates. The problem sets covered simplifying logical expressions (seven problems) and transforming logical expressions to the negation normal form (four problems), respectively. Students worked on both problem sets while thinking aloud in a fixed sequence until the time ran out. This decision was based on both problem types having increasing difficulty levels, with the first problems set including additional problem-solving scaffolds via reason boxes.

3.3 Data and Coding SRL Categories

The dataset analyzed comprised individual student transactions encoded in log data (e.g., attempts, hint requests) from all three tutoring systems, along with think-aloud transcripts. Student actions within each ITS were logged into PSLC DataShop [34]. Whisper, an open-source transcription model for voice, generated think-aloud transcripts, which segmented utterances with start and end timestamps. Error and accuracy reports are discussed in [50].

In the current study, we merged multiple utterances falling between the same timestamped student transactions, which allows modeling the subsequent student action's correctness based on prior utterances' SRL codes, as done in [12]. Synchronization of log data and think-aloud transcriptions was ensured by coding a reference tutor transaction by a coder familiar with the software, which allows for synchronization with no more than a 1-second error margin. Log data and anonymized synchronized think-aloud transcripts are available at (https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=5371) for the Stoichiometry and ORCCA tutor, and at (https://pslcdatashop.web.cmu.edu/DatasetInfo?datasetId=5820) for the Logic Tutor.

Concatenated utterances were annotated following a coding scheme (see Table 1) aligning with the four-stage SRL model by Winne and Hadwin [60]: *Processing Information, Planning, Enacting*, and *Realizing Errors*. These categories, focusing on relevant behaviors within problem-solving learning environments, represent a subset of SRL behaviors within each model stage. Although coarser-grained than other SRL think-aloud studies, this approach allows observation of finer-grained cognitive operations within comparatively short utterances between problem-solving attempts.

The coding categories reflect critical behaviors at different stages of problem-solving learning, allowing us to examine learners' cognitive activities during information processing, planning, enacting conceptual actions, and realizing errors. Coding at this level enables observation of cognitive operations that are usually inferred from multiple actions and verbalizations. Table 1 outlines the coding categories and related behaviors.

Table 1. the four SRL categories, including indicative behaviors of each category and example utterances.

SRL Category	Behavior	Example Utterance
Processing Information	• Assemble information The utterance demonstrates behaviors where students read or re-read a question, hints, or feedback provided by the system • Comprehend information The utterance demonstrates behaviors where students repeat information provided by the system with a level of synthesis	"Let's figure out how many hydrogen items are in a millimole of water molecule H2O molecules. Our result should have three significant features. Figures. Avogadro number is 6.02 E23. 2 atoms of H2O."
Planning	• Identify goals and form plans The utterance reflects behaviors where students verbalize a conceptual plan of how they will solve the problem	"Our goal of the result is hydrogen atoms. The goal of the result is the number of hydrogen atoms, right?"
Enacting	• Verbalize previous action The utterance reflects students' behaviors where they verbalize an action that has just been carried out explaining what they did • Announce the next action The utterance reflects student behaviors where they verbalize a concrete and specific action that they will do next	"Two molecules of this. How many atoms in a... How many atoms in a minimum molecule of M mole? 61023 divided by 2. 3.0115."
Realizing Errors	• Realize something is wrong The utterance demonstrates instances where students realize there is a mistake in the	"It's incorrect. What's happened? It is the thousand in the wrong spot. 32 grams per mole. No, the thousand is

	answer or the process with or without external prompting (i.e., tutor feedback)	correct, so what am I doing wrong? [...]"

Two coders established acceptable inter-rater reliability after coding 162 concatenated utterances ($K_{processing}$= 0.78, $K_{planning}$ = 0.90,$K_{enacting}$ = 0.77, K_{errors} = 1.00). They then individually coded the remaining utterances, double-coding any lacking agreement within the inter-rater iteration. The present study sampled a total of 955 annotated utterances. We reported the distribution of the codes of each SRL category observed in each platform in Table 2.

Table 2. the distribution of the SRL Codes in each platform

	Stoichiometry	ORCCA	Logic Tutor
Total number of students	9	5	5
Total number of labeled utterances	469	162	324
Processing Information	13%	21%	26%
Planning	11%	12%	17%
Enacting	23%	19%	31%
Realizing Errors	6%	4%	19%

3.4 Sentence Embedding and Building Detectors

With the goal of developing models that can reliably detect the presence or absence of the four SRL categories (*Processing Information*, *Planning*, *Enacting*, *Realizing Errors*), we trained machine learning models using features obtained from two state-of-the-art sentence embeddings models, and compared their performance. Specifically, we used the Universal Sentence Encoder large v5 (USE; [13]) and the "text-embedding-3-small" model from OpenAI [45] to vectorize the utterances.

These sentence embedding models operate by encoding sentences into a high-dimensional vector space, utilizing the relationships among words learned from previously encountered sentences. This approach enables the models to generate context-aware representations of words, taking into account both the ordering and identity of all other words in the text. Through the process of vectorization, these sentence embedding models utilize a series of neural networks to convert the text-based input into a numerical representation as output.

The two models employed in the current study differ in the length of embeddings they produce: the length of the embedding vector is 512 for the USE and 1536 for the "text-embedding-3-small". In other words, through the process of vectorization, each clip (concatenated utterances) was converted to a vector that contains 512 numerical values using the USE model and 1536 numerical values using the "text-embedding-3-small" model. The dimensionality of the vector is important for downstream prediction tasks as it encodes varying amounts of information. However, higher dimensions may not always be optimal, as it may potentially include irrelevant information for the downstream prediction task and could be memory-intensive [55]. Given the limited work on examining the impact of dimensionality on model performance in this particular scenario – predicting students' SRL processes in think-aloud – we explored and compared two pre-trained embedding models with their default vector size.

These numerical values derived from the embedding models were then used as features to train machine learning (ML) models that predict the presence or absence of the four SRL categories. These ML models were fitted using a neural network with one hidden layer, using the TensorFlow library in Python. (see Github repository for Python script - https://github.com/pcla-code/EDM24_SRL-detectors-for-think-aloud.git)

To evaluate the model performance, we employed 5-fold student-level cross-validation. This validation method involved splitting all clips (concatenated utterances) into five folds, with each student's clips nested in one-fold. For training the model, four folds were used, while the fifth fold served as the test set for predictions. Each fold acted as the test set once. By splitting the data at the student level, we ensured that no two clips from the same student could be included in both the training and testing sets, thus preventing data leakage that would bias model results toward higher predictive accuracy. This approach enhances the model's ability to generalize to new students and is recommended as the standard practice to validate ML models in the field of educational data mining [6]. For each test set, we computed the area under the Receiver Operating Characteristic curve (AUC), and then averaged them across the five test sets, reporting an average AUC for each model.

To compare the performance difference between the two embedding methods, we computed the overall AUC using the predictions compiled from the five test sets for each model. We used DeLong's test to evaluate whether the difference in the overall AUCs between the two embedding methods is significant for each prediction task.

3.5 Examine the Transferability of the Models

We investigated the generalizability of our model across domain (stoichiometry chemistry to formal logic) and ITS interface types (open-ended to highly scaffolded). Building platform- and domain-independent models has been a long-standing research goal in learning analytics and educational data mining, as generalizable models do not require new training data in new contexts. However, past work has also shown that domain transfer might depend on the similarity of tutoring system interfaces and architectures [49].

To investigate transfer, a natural split between the three ITS in this study can be made by domain (training on Stoichiometry Tutor and ORCCA, testing on Logic Tutor) and platform type (training on the open-ended ORCCA and Logic Tutor, testing on the highly-scaffolded Stoichiometry Tutor). We evaluated model performance on these two test-sets analogous to our cross-validation procedure described above, reporting test set AUC.

3.6 Error Analysis

To better understand the performance and limitations of our employed models we performed an informal error analysis of misclassified examples at the two transfer tasks: subject and platform type transfer. The goal of the error analysis is to understand the nature of the errors made by the model when it is applied to a new domain or platform. This helps in identifying patterns or systematic mistakes, which can lead to insights on how to improve the model's architecture [20].

The error analysis followed a multi-stage process. One research team member categorized all misclassified data points by (a) label and (b) error type (false positive vs. false negative). Then, the same coder categorized errors within these groups by an initial set of error types and themes. Afterward, a second research team member consolidated these themes identified in errors and revised their description through discussion with the first coder. This process minimized bias in individual coders. Given the informal nature of this analysis, we report the most interesting error themes and patterns that can point to further model improvements in future research.

4. RESULTS

4.1 Model Performance

Table 3 reports the average and standard deviation of AUC scores by SRL category obtained from 5-fold student-level cross-validation with two embedding methods. The AUC indicates the probability that the model can correctly classify a set of one positive and one negative example of each class. An AUC of 0.5 corresponds to classification at random chance, while an AUC of 1 represents perfect classification.

As the results suggested, the models performed fairly well in predicting the four SRL categories, showing average AUCs ranging from 0.696 to 0.915. These findings suggest that the detectors were generally successful in capturing the four SRL categories, using either embedding method. We also note that *Realizing Errors* was more accurately detected than the other three SRL categories. This could possibly be due to the use of signifying words or phrases in student verbalizations that are indicative of this category, such as "wrong" or "it's incorrect."

To evaluate whether the models using the two different embedding methods differ significantly in performance, DeLong's test was conducted. For each SRL category, we first computed the overall AUC using the predictions compiled from the five test sets for each embedding method. DeLong's test was then applied to examine if the difference in the overall AUCs is significant. As shown in Table 3, we found that models built using the embeddings from text-embedding-3-small significantly and consistently outperformed the models built using the embeddings from USE for all four SRL categories.

Table 3. Average AUC and DeLong's test results: comparing the model performance between the two embedding models

	Avg. AUC w/ USE	Avg. AUC w/ text-embedding-3-small	Z-score	95% CI	p
Processing Information	.793 (.029)	.828 (.038)	-2.57	[-0.07, -0.01]	.010
Planning	.716 (.042)	.785 (.023)	-3.30	[-0.11, -0.03]	<0.001
Enacting	.696 (.061)	.779 (.076)	-4.18	[-0.11, -0.04]	<0.001
Realizing Errors	.865 (.021)	.915 (.031)	-2.27	[-0.08 0.01]	.02

4.2 Transferability of the Models

4.2.1 Transfer across subject areas

To evaluate if models can transfer across subject areas, we trained models using data from Stoichiometry and ORCCA tutors (both cover stoichiometry chemistry) and tested the models on data collected from Logic Tutor. As shown in Table 4, we found, models using either set of embeddings (USE or text-embedding-3-short) were lacking the ability to predict three of the four SRL categories when transferred across subject areas. Specifically, when models were trained on the subject of chemistry, they did not transfer well predicting when students were *Processing Information*, *Planning*, and *Enacting* while working on formal logic questions, with AUC scores ranging from 0.558 to 0.654. However, models that predict *Realizing Errors* did transfer across subject areas.

Table 4. AUC tested on Logic Tutor with models trained on Stoichiometry and ORCCA Tutor: examine model transfer across *subject areas*

SRL category	USE	text-embedding-3-short
Processing Information	0.654	0.593
Planning	0.558	0.619
Enacting	0.561	0.605
Realizing Errors	0.784	0.896

To better understand the model's transferability across domains (from stoichiometry chemistry to formal logic), which could inform future model refinements, we manually inspect classification examples across all four SRL categories. We highlight three notable patterns, based on discussions among two research team members.

Firstly, by examining the errors in the *Processing* and Planning categories, we noticed that the model prediction often coincided with domain-specific vocabularies. *Processing* encodes verbalizations where students read instructions or hints provided by the system, through which students obtain and comprehend information. However, the language used in instruction (e.g., the wording of a question or hints) is domain-specific, which potentially causes issues in transferring the prediction to a different domain. For example, in a misclassified example, a student said, "This is the same in both expressions./I see that./That means we have associativity". In this case, we observe that the student was processing what they noticed, making a comparison between two expressions and comprehending the problem; however, given the use of domain-specific vocabularies (e.g., "expressions", "associativity") that is specific to Logic Tutor, the model lacked the ability to understand that the student was processing information, which resulted in a misclassification. A similar pattern was observed in the *Planning* category.

Secondly, when detecting *Enacting* behavior, the model was more likely to be accurate if the utterances included general actions words, such as "put", "try", "going to", and "enter". For example, a correctly classified example included "OK, so I'm just going to enter what this says, the answer should be Q, P". However, when the utterances did not include an action word and the student used language that is specific to formal logic, the model tended to have difficulty inferring the *Enacting* category, and classifying it inaccurately. For instance, a student said, "not q and p/ okay further solve p or parenthesis not q and p with absorption". In this example, the student was describing what they just entered in the formula in Logic Tutor, in which they entered the negation symbol (!) and the "and" symbol (&) along with the two parameters p and q. As we

see, the words “not” and “and " in the utterance here imply logical operators rather than how they are typically used in daily language. Because of the use of vocabulary that has special meaning in this context that involves formal logic, the model failed to understand the scenario in which the student was describing an entry, representing an *Enacting* behavior.

Finally, we noticed that *Realizing Errors* models transfer fairly well across domains, which may be due to the use of high-signal words, such as “no”, “wrong”, and “oh”, which are domain-agnostic. However, false negative errors were still prevalent in predictions. Potential issues in the misclassification could stem from the dual meaning of “negation” and “negative” in formal logic (which is, again, domain-specific language distinct to Logic Tutor). For example: "So this would be negation of q and the junction of negative p./And this seems to be incorrect” which the model incorrectly classified as not *Realizing Error* (false negative). In other words, given that “not” and “negation” have domain-relevance in logic rather than being common in error-making, the model likely misclassified this category during transfer.

4.2.2 Transfer across platform design

To assess whether models can transfer across platform designs, we trained models using data collected from ORCCA and Logic Tutor (both of which use an interface design with dynamic scaffolding and open-ended response formula entry fields) and tested the models on Stoichiometry Tutor, which features a highly-scaffolded interface. In Table 5, we report the AUC of these models predicting the utterances in the test set. The results suggest that these models are successful at transferring across designs, as indicated by AUC scores ranging from 0.713 to 0.882.

Table 5. AUC tested on Stoichiometry Tutor with models trained on ORCCA and Logic Tutor: examine model transfer across *platform design*

SRL category	USE	text-embedding-3-short
Processing Information	0.816	0.846
Planning	0.814	0.882
Enacting	0.713	0.808
Realizing Errors	0.81	0.849

To further examine these models’ transferability across platform designs, we manually inspect misclassified examples across all four SRL categories. Through this process, we note two error patterns that might be caused by the differences in platform design.

Firstly, we noticed that multiple utterances were incorrectly tagged with the *Planning* category, but were actually an act of *Processing*, as participants read or summarized instructions in Stoichiometry Tutor. Specifically, error-specific feedback in Stoichiometry Tutor guides students’ problem-solving and planning through hints, which often start with language like “Isn’t our goal to [...]”. The use of goal-oriented language in system information might have confused the model, making it believe that a student was forming goals, while in reality it was the student reading off of hint/feedback, which constitutes *Processing Information*.

Additionally, we noticed that the model tended to misclassify *Enacting* behavior in Stoichiometry Tutor, and that is possibly due to a step where students use a drop-down menu to self-explain, a step that is unique to Stoichiometry Tutor among the three ITSs. In this step, students label the rationale behind their problem-solving steps. For example, they may select "unit conversion" in the drop-down menu to explain their process of converting grams to kilograms. Consequently, when a student verbalizes "Unit conversion from grams to kilograms" immediately after selecting "unit conversion" in the drop-down menu, the model misclassifies this utterance as *Planning* rather than *Enacting*. This misclassification occurs potentially because the model was not trained on data that captures this specific step of the question or behavioral log data that reflects the student's action.

5. DISCUSSION

5.1 Main Findings and Contributions

Think-aloud protocols (TAPs) are an important approach for investigating SRL during problem-solving and have been widely used in prior research [9, 26, 28, 37]. Previous studies not only demonstrate TAPs as a valid approach to measure SRL but also underlines its importance in capturing SRL in a more comprehensive way, complementing behavioral log data in SRL measurement [19]. However, TAPs require researchers to manually code students' verbalizations, which presents challenges in scalability. Thus, the present study demonstrated the feasibility of using large language models and machine learning to automatically identify SRL behaviors in machine-generated student think-aloud transcripts, which can speed research employing TAPs. Specifically, we collected students’ think-aloud while working within three intelligent tutoring systems. We vectorized students’ utterances using two different sentence embedding models (Universal Sentence Encoders and OpenAI’s text-embedding-3-short), and then used the embeddings as features to train machine learning models that predict four SRL categories (i.e., *Processing Information, Planning, Enacting, and Realizing Errors*). The transferability of these models was evaluated across subject areas and platform designs. Our main findings and contributions are as follows.

First, by evaluating the models using 5-fold student-level cross-validation, we demonstrated that embeddings from either Universal Sentence Encoders or OpenAI’s text-embedding-3-short were reliable at detecting the four SRL categories, with average AUC scores ranging from 0.696 to 0.915. These results demonstrate a promising potential of leveraging these models to measure students’ SRL in think-aloud protocols without human supervision. Additionally, when comparing the two embedding models, we found that models utilizing embeddings from OpenAI’s text-embedding-3-short performed significantly better than Universal Sentence Encoders. This advantage in performance may be partially attributed to the fact that text-embedding-3-short generates a larger vector, potentially extracting and retaining more relevant information for the prediction task, thus resulting in a better performance [55].

Second, by training models using data from two of the three platforms and testing them on the third platform, we examined the transferability of these models across subject areas, which prior work on detectors in educational data mining identified as challenging [7]. Specifically, we found that these models lack the ability to transfer across domains (i.e., from stoichiometry chemistry to formal logic), showing subpar performance in predicting all SRL categories other than *Realizing Errors*. Upon examining the misclassified cases across the four SRL categories, we noticed three error patterns, mainly relating to the use of domain-specific language. Specifically, we noticed that these

models rely heavily on domain-specific vocabularies when predicting the *Processing Information* and *Planning* categories. When students use vocabulary that is specific to formal logic which is unlikely to occur in the domain of chemistry (e.g., "associative" or "DeMorgan rule"), the models tend to make false negative errors. Additionally, when detecting *Enacting*, the model tends to confuse the meaning of logical operators (e.g., "not" or "and") transcribed to natural language instead of their conventional symbols (i.e., "~" and "&") with their meaning in everyday language. Thus, when students verbalize their formula entry in Logic Tutor (e.g., students entered ~q&p in the formula and verbalized "not q and p"), the model fails to recognize it as *Enacting*. Similarly, because of the dual meaning of "negation" and "negative" in formal logic (which is, again, domain-specific language specific to Logic Tutor), the *Realizing Error* model which was not trained on the subject of formal logic failed to transfer, potentially failing to understand that "not" and "negation" have domain relevance in formal logic questions. Given these observations, in the next section, we discuss a potential improvement of including domain-specific language in model training.

Third, using the same approach (training models on data from two platforms with similar design and testing the models on the third platform), we show that these models can transfer across platforms, demonstrating a successful transferability from open-ended, formula-based platforms to a highly structured platform with fraction-based input. Similar to [49], despite showing a success of transfer of these ML models across platforms, we noticed that the success of transfer depends on how close or different the design features are among platforms. By examining the misclassified utterances closely, we found two error patterns that relate to the differences in platform design that may contribute to the misclassification. Specifically, we found that the feedback/error message in Stoichiometry Tutor contains language that is different from the error messages used in the other two platforms (ORCCA and Logic Tutor). Specifically, the error message in Stoichiometry Tutor contains language (e.g., "Isn't our goal to…") that insinuates that a student is making a plan and setting goals while, in actuality, the student is reading the error message. Because of this, the models were likely to classify these instances as *Planning* rather than *Processing*. Additionally, we found the step where students use a drop-down in Stoichiometry Tutor to self-explain (a step that is used in Stoichiometry Tutor but not in the other two systems) may be the reason why the model misclassified *Enacting* for *Planning*. In these cases, students verbalize their selection (an indication of enactment in the context of selecting options in drop-downs); however, without the context of the interface and not knowing a student's prior and subsequent actions, the model can easily confuse between the two scenarios when the student said "unit conversion": [I choose "unit conversion" as a reason in the drop-down menu] vs. [I'm going to do a "unit conversion"], where the former is *Enacting* while the latter is *Planning*. This misclassification occurs potentially because the model was not trained on data that captures this specific step of the question or behavioral log data that reflects the student's action. We discuss the incorporation of behavioral logs as a potential solution to improve models' transferability in the next section.

5.2 Limitations and Future Work

We acknowledge several limitations of the current work regarding the use of data and the choice of models. These limitations should be addressed and further evaluated in future work.

First, students' utterances (i.e., transcriptions of what students said while thinking aloud) were the main source of data to train models and predict SRL behaviors. However, this approach may potentially overlook contextual information in the learning environment, which is not reflected in students' verbalizations. Our error analysis suggests that missing contextual information (e.g., problem statements, hints, and feedback messages) containing domain and platform-specific language limits model transfer across subject areas and platforms. This result aligns with recent research that identifies limitations in classifying tutorial dialogue independent of its problem-solving context [11]. Future work could evaluate model transfer of SRL prediction tasks when incorporating instructional context. Additionally, model transfer could be improved by considering learning relations between vocabulary of domains and platforms to one another. Past approaches include learning a linear mapping from representations (e.g., embeddings) of one domain to another, which has been successfully used in preserving relationships between modalities in machine learning research [41]. Finally, this work could enable investigation into whether certain domains transfer more effectively to one another than others. For example, ITS in STEM domains might transfer better to one another than STEM domains due to vocabulary learning and non-STEM problem-solving contexts [32].

Second, upon analyzing the misclassified cases, we observed a potential advantage in integrating behavioral logs into model training. These logs, used in human coding, assisted coders in contextualizing their assessment of a verbalization within the context of sequential actions; however, they were not utilized in model training. While the models demonstrated satisfactory performance without behavioral logs, enriching the dataset to include such logs may enhance transferability, particularly as these behaviors can be specific to the platform being used.

Third, we recognize a potential limitation with the use of auto-transcribing tools, in which the meaning of domain-specific words and language may not be correctly captured based on the context. For example, the transcription of "not" when students were verbalizing the negation symbol (~). Therefore, it may be appropriate for future projects along this line to manually review and revise transcripts and examine how it affects models' performance and their ability to transfer.

Fourth, the current study experimented with two state-of-the-art embedding models and one ML algorithm to predict SRL in think-aloud. Although we demonstrate the success of these models using the current approach, future study may take other factors, such as cost, time, model efficiency into consideration, when considering the scalability of these models. For example, although neural networks have been considered as the default and preferred algorithm for prediction tasks that process natural language [22], a recent study demonstrates a success of using random forest, a more computationally effective model, to examine students' reading comprehension in think-aloud [61]. For providing SRL measurement at scale, efficiency or cost-effectiveness analysis might be a necessary next step in this line of research.

The future of work along these lines is likely to be impacted by the emerging use of ChatGPT for scalable coding of qualitative data (e.g., [30, 62]). In these studies, prompts are given to ChatGPT to instruct coding, along with definitions and examples. Compared to the current method, coding with ChatGPT can be faster, as it does not directly involve vectorizing text and training a machine learning model. Although [30] found that ChatGPT with prompt engineering can be less accurate than classical NLP models for

some problems, it is worth investigating the possibility of coding think-aloud protocols using ChatGPT. There is a possibility that directly applying ChatGPT for classification might perform better than our approach in specific classification tasks where we observed comparatively low accuracy (e.g., domain transfer). Future research should explore the question of when the current method (sentence embedding and machine learning) is preferred over prompt-based large language models.

6. CONCLUSION

The present study scaled up the measurement of SRL in students' think-aloud through automated transcription and LLM-based prediction. Researchers can leverage our methodology to expedite research based on think-aloud protocols, including analyzing SRL behaviors across contexts at scale. We established reliable models to detect four SRL categories, identifying the key behaviors in each stage of the Winne and Hadwin four-stage model. These models successfully transfer across tutoring systems of the same domain with different interfaces. However, our results also suggest that model transfer to new learning environments can likely be improved by incorporating linguistic information found in interface instructions, including problem statements, dropdown items, and as-needed instructions in the form of error feedback and hints. Such language could be combined with student verbalizations into a joint embedding. Similarly, learning a domain-general SRL model based solely on student verbalizations is challenging, as a model trained on a chemistry domain context did not generalize to a tutoring system for formal logic. More work is needed to address domain-specific vocabulary in prediction, a common source of error in our models, including learning linear transformations of embedding spaces from different domains. Overall, this study motivates further inquiry into automated analysis of SRL from natural language and contributes to the emerging field of advanced natural language models in educational data mining and intelligent tutoring systems.

7. ACKNOWLEDGEMENTS

Carnegie Mellon University's GSA/Provost GuSH Grant funding was used to support this research.

8. REFERENCES

[1] Aleven, V., McLaren, B., Roll, I. and Koedinger, K. 2006. Toward Meta-cognitive Tutoring: A Model of Help Seeking with a Cognitive Tutor. *International Journal of Artificial Intelligence in Education*. 16, 2 (2006), 101–128.

[2] Aleven, V., McLaren, B.M., Sewall, J., Van Velsen, M., Popescu, O., Demi, S., Ringenberg, M. and Koedinger, K.R. 2016. Example-Tracing Tutors: Intelligent Tutor Development for Non-programmers. *International Journal of Artificial Intelligence in Education*. 26, 1 (Mar. 2016), 224–269. DOI:https://doi.org/10.1007/s40593-015-0088-2.

[3] Araka, E., Maina, E., Gitonga, R. and Oboko, R. 2020. Research trends in measurement and intervention tools for self-regulated learning for e-learning environments—systematic review (2008–2018). *Research and Practice in Technology Enhanced Learning*. 15, 1 (Dec. 2020), 6. DOI:https://doi.org/10.1186/s41039-020-00129-5.

[4] Azevedo, R., Johnson, A., Chauncey, A., Graesser, A.C., Zimmerman, B. and Schunk, D. 2011. Use of hypermedia to assess and convey self-regulated learning. *Handbook of Self-Regulation of Learning and Performance*. 32, (2011), 102–121.

[5] Azevedo, R., Taub, M. and Mudrick, N.V. 2017. Understanding and Reasoning about Real-Time Cognitive, Affective, and Metacognitive Processes to Foster Self-Regulation with Advanced Learning Technologies. *Handbook of Self-Regulation of Learning and Performance*. (2017), 254–270.

[6] Baker, R.S. 2023. Big Data and Education.

[7] Baker, R.S. 2019. Challenges for the Future of Educational Data Mining: The Baker Learning Analytics Prizes. 11, 1 (2019), 17.

[8] Baker, R.S., Corbett, A.T. and Koedinger, K.R. 2004. Detecting student misuse of intelligent tutoring systems. *International Conference on Intelligent Tutoring Systems*. (2004), 531–540.

[9] Bannert, M., Reimann, P. and Sonnenberg, C. 2014. Process mining techniques for analysing patterns and strategies in students' self-regulated learning. *Metacognition and Learning*. 9, 2 (Aug. 2014), 161–185. DOI:https://doi.org/10.1007/s11409-013-9107-6.

[10] Biswas, G., Jeong, H., Kinnebrew, J.S., Sulcer, B. and Roscoe, R. 2010. Measuring Self-regulated Learning Skills Through Social Interactions in a Teachable Agent. *Research and Practice in Technology Enhanced Learning*. 05, 02 (2010), 123–152.

[11] Borchers, C., Yang, K., Lin, J., Rummel, N., Koedinger, K. and Aleven, V. 2024. Combining Dialog Acts and Skill Modeling: What Chat Interactions Enhance Learning Rates During AI-Supported Peer Tutoring? *Proceedings of the 17th International Conference on Educational Data Mining* (2024).

[12] Borchers, C., Zhang, J., Baker, R.S. and Aleven, V. 2024. Using Think-Aloud Data to Understand Relations between Self-Regulation Cycle Characteristics and Student Performance in Intelligent Tutoring Systems. *Proceedings of the 14th International Conference on Learning Analytics and Knowledge* (2024).

[13] Cer, D., Yang, Y., Kong, S., Hua, N., Limtiaco, N., John, R.S., Constant, N., Guajardo-Cespedes, M., Yuan, S., Tar, C., Sung, Y.-H., Strope, B. and Kurzweil, R. 2018. Universal Sentence Encoder. arXiv:1803.11175.

[14] Cleary, T.J. and Chen, P.P. 2009. Self-regulation, motivation, and math achievement in middle school: Variations across grade level and math context. *Journal of School Psychology*. 47, 5 (2009), 291–314.

[15] Crossley, S.A., Louwerse, M.M., McCarthy, P.M. and McNamara, D.S. 2007. A Linguistic Analysis of Simplified and Authentic Texts. *The Modern Language Journal*. 91, 1 (2007), 15–30.

[16] Darvishi, A., Khosravi, H., Sadiq, S. and Gašević, D. 2022. Incorporating AI and learning analytics to build trustworthy peer assessment systems. *British Journal of Educational Technology*. 53, 4 (Jul. 2022), 844–875. DOI:https://doi.org/10.1111/bjet.13233.

[17] Deekens, V.M., Greene, J.A. and Lobczowski, N.G. 2018. Monitoring and depth of strategy use in computer-based learning environments for science and history. *British Journal of Educational Psychology*. 88, 1 (Mar. 2018), 63–79. DOI:https://doi.org/10.1111/bjep.12174.

[18] Ericsson, K.A. and Simon, H.A. 1998. How to Study Thinking in Everyday Life: Contrasting Think-Aloud Protocols With Descriptions and Explanations of Thinking. *Mind, Culture, and Activity*. 5, 3 (Jul. 1998), 178–186. DOI:https://doi.org/10.1207/s15327884mca0503_3.

[19] Fan, Y., Rakovic, M., van der Graaf, J., Lim, L., Singh, S., Moore, J., Molenaar, I., Bannert, M. and Gašević, D. 2023. Towards a fuller picture: Triangulation and integration of

the measurement of self-regulated learning based on trace and think aloud data. *Journal of Computer Assisted Learning*. 39, 4 (Aug. 2023), 1303–1324. DOI:https://doi.org/10.1111/jcal.12801.

[20] Feng, M. 2005. Looking for Sources of Error in Predicting Student's Knowledge. *In Educational data mining: Papers from the 2005 AAAI workshop* (2005), 54–61.

[21] Fox, M.C., Ericsson, K.A. and Best, R. 2011. Do procedures for verbal reporting of thinking have to be reactive? A meta-analysis and recommendations for best reporting methods. *Psychological Bulletin*. 137, 2 (Mar. 2011), 316–344. DOI:https://doi.org/10.1037/a0021663.

[22] Goldberg, Y. 2016. A Primer on Neural Network Models for Natural Language Processing. *Journal of Artificial Intelligence Research*. 57, (Nov. 2016), 345–420. DOI:https://doi.org/10.1613/jair.4992.

[23] van der Graaf, J., Lim, L., Fan, Y., Kilgour, J., Moore, J., Gašević, D., Bannert, M. and Molenaar, I. 2022. The Dynamics Between Self-Regulated Learning and Learning Outcomes: an Exploratory Approach and Implications. *Metacognition and Learning*. 17, 3 (Dec. 2022), 745–771. DOI:https://doi.org/10.1007/s11409-022-09308-9.

[24] Greene, J.A. and Azevedo, R. 2010. The Measurement of Learners' Self-Regulated Cognitive and Metacognitive Processes While Using Computer-Based Learning Environments. *Educational Psychologist*. 45, 4 (Oct. 2010), 203–209.

[25] Greene, J.A., Deekens, V.M., Copeland, D.Z. and Yu, S. 2017. Capturing and Modeling Self-Regulated Learning Using Think-Aloud Protocols. *Handbook of Self-Regulation of Learning and Performance*. D.H. Schunk and J.A. Greene, eds. Routledge. 323–337.

[26] Greene, J.A., Robertson, J. and Costa, L.-J.C. 2011. Assessing Self-Regulated Learning Using Think-Aloud Methods. *Handbook of Self-Regulation of Learning and Performance*. 313–328.

[27] Hatala, M., Nazeri, S. and Salehian Kia, F. 2023. Progression of students' SRL processes in subsequent programming problem-solving tasks and its association with tasks outcomes. *The Internet and Higher Education*. 56, (Jan. 2023), 100881. DOI:https://doi.org/10.1016/j.iheduc.2022.100881.

[28] Heirweg, S., De Smul, M., Devos, G. and Van Keer, H. 2019. Profiling upper primary school students' self-regulated learning through self-report questionnaires and think-aloud protocol analysis. *Learning and Individual Differences*. 70, (Feb. 2019), 155–168. DOI:https://doi.org/10.1016/j.lindif.2019.02.001.

[29] Heirweg, S., De Smul, M., Merchie, E., Devos, G. and Van Keer, H. 2020. Mine the process: investigating the cyclical nature of upper primary school students' self-regulated learning. *Instructional Science*. 48, 4 (Aug. 2020), 337–369. DOI:https://doi.org/10.1007/s11251-020-09519-0.

[30] Hutt, S., DePiro, A., Wang, J., Rhodes, S., Baker, R.S., Hieb, G., Sethuraman, S., Ocumpaugh, J. and Mills, C. 2024. Feedback on Feedback: Comparing Classic Natural Language Processing and Generative AI to Evaluate Peer Feedback. *Proceedings of the 14th Learning Analytics and Knowledge Conference* (Kyoto Japan, 2024), 55–65.

[31] Hutt, S., Ocumpaugh, J., Biswas, G. and Baker, R.S. 2021. Investigating SMART Models of Self-Regulation and their Impact on Learning. *Proceedings of the 14th International Conference on Educational Data Mining*. (2021), 580–587.

[32] Kazi, S.A. 2005. VocaTest: An Intelligent Tutoring System for Vocabulary Learning using the "mLearning" Approach. (2005).

[33] King, E.C., Benson, M., Raysor, S., Holme, T.A., Sewall, J., Koedinger, K.R., Aleven, V. and Yaron, D.J. 2022. The Open-Response Chemistry Cognitive Assistance Tutor System: Development and Implementation. *Journal of Chemical Education*. 99, 2 (Feb. 2022), 546–552. DOI:https://doi.org/10.1021/acs.jchemed.1c00947.

[34] Koedinger, K.R., Leber, B. and Stamper, J. 2010. A Data Repository for the EDM community: The PSLC DataShop. *Handbook of educational data mining*. 43–56.

[35] Kovanović, V., Joksimović, S., Mirriahi, N., Blaine, E., Gašević, D., Siemens, G. and Dawson, S. 2018. Understand students' self-reflections through learning analytics. *Proceedings of the 8th International Conference on Learning Analytics and Knowledge* (Sydney New South Wales Australia, Mar. 2018), 389–398.

[36] Labuhn, A.S., Zimmerman, B.J. and Hasselhorn, M. 2010. Enhancing students' self-regulation and mathematics performance: the influence of feedback and self-evaluative standards. *Metacognition and Learning*. 5, 2 (2010), 173–194.

[37] Lim, L., Bannert, M., van der Graaf, J., Molenaar, I., Fan, Y., Kilgour, J., Moore, J. and Gašević, D. 2021. Temporal Assessment of Self-Regulated Learning by Mining Students' Think-Aloud Protocols. *Frontiers in Psychology*. 12, (Nov. 2021), 1–18. DOI:https://doi.org/10.3389/fpsyg.2021.749749.

[38] McLaren, B.M., DeLeeuw, K.E. and Mayer, R.E. 2011. Polite web-based intelligent tutors: Can they improve learning in classrooms? *Computers & Education*. 56, 3 (Apr. 2011), 574–584. DOI:https://doi.org/10.1016/j.compedu.2010.09.019.

[39] McLaren, B.M., Lim, S.-J., Gagnon, F., Yaron, D. and Koedinger, K.R. 2006. Studying the Effects of Personalized Language and Worked Examples in the Context of a Web-Based Intelligent Tutor. *Intelligent Tutoring Systems*. M. Ikeda, K.D. Ashley, and T.-W. Chan, eds. Springer Berlin Heidelberg. 318–328.

[40] McNamara, D., Allen, D.S., Crossley, S.A., Dascalu, M. and Perret, C.A. 2017. Natural language processing and learning analytics. *Handbook of learning analytics*. 93–104.

[41] Merullo, J., Castricato, L., Eickhoff, C. and Pavlick, E. 2023. Linearly Mapping from Image to Text Space. arXiv:2209.15162.

[42] Molenaar, I., Horvers, A. and Baker, R.S. 2021. What can moment-by-moment learning curves tell about students' self-regulated learning? *Learning and Instruction*. 72, (Apr. 2021), 101206. DOI:https://doi.org/10.1016/j.learninstruc.2019.05.003.

[43] Moos, D.C. and Azevedo, R. 2008. Self-regulated learning with hypermedia: The role of prior domain knowledge. *Contemporary Educational Psychology*. 33, 2 (Apr. 2008), 270–298. DOI:https://doi.org/10.1016/j.cedpsych.2007.03.001.

[44] Nasiar, N., Baker, R.S., Zou, Y., Zhang, J. and Hutt, S. 2023. Modeling Problem-Solving Strategy Invention (PSSI) Behavior in an Online Math Environment. *Artificial Intelligence in Education. Posters and Late Breaking Results, Workshops and Tutorials, Industry and Innovation Tracks, Practitioners, Doctoral Consortium and Blue Sky*. N. Wang, G. Rebolledo-Mendez, V. Dimitrova, N. Matsuda,

and O.C. Santos, eds. Springer Nature Switzerland. 453–459.

[45] Neelakantan, A. et al. 2022. Text and Code Embeddings by Contrastive Pre-Training. arXiv:2201.10005.

[46] Nota, L., Soresi, S. and Zimmerman, B.J. 2004. Self-regulation and academic achievement and resilience: A longitudinal study. *International Journal of Educational Research*. 41, 3 (2004), 198–215.

[47] Panadero, E. 2017. A Review of Self-regulated Learning: Six Models and Four Directions for Research. *Frontiers in Psychology*. 8, (2017), 422.

[48] Paquette, L. and Baker, R.S. 2019. Comparing machine learning to knowledge engineering for student behavior modeling: a case study in gaming the system. *Interactive Learning Environments*. 27, 5–6 (2019), 585–597.

[49] Paquette, L., Baker, R.S., de Carvalho, A. and Ocumpaugh, J. 2015. Cross-System Transfer of Machine Learned and Knowledge Engineered Models of Gaming the System. *User Modeling, Adaptation and Personalization*. F. Ricci, K. Bontcheva, O. Conlan, and S. Lawless, eds. Springer International Publishing. 183–194.

[50] Radford, A., Kim, J.W., Xu, T., Brockman, G., McLeavey, C. and Sutskever, I. 2023. Robust Speech Recognition via Large-Scale Weak Supervision. *International Conference on Machine Learning* (2023), 28492–28518.

[51] Saint, J., Fan, Y., Gašević, D. and Pardo, A. 2022. Temporally-focused analytics of self-regulated learning: A systematic review of literature. *Computers and Education: Artificial Intelligence*. 3, (2022), 100060. DOI:https://doi.org/10.1016/j.caeai.2022.100060.

[52] Saint, J., Whitelock-Wainwright, A., Gasevic, D. and Pardo, A. 2020. Trace-SRL: A Framework for Analysis of Microlevel Processes of Self-Regulated Learning From Trace Data. *IEEE Transactions on Learning Technologies*. 13, 4 (Oct. 2020), 861–877. DOI:https://doi.org/10.1109/TLT.2020.3027496.

[53] Schooler, J.W., Ohlsson, S. and Brooks, K. 1993. Thoughts Beyond Words: When Language Overshadows Insight. *Journal of Experimental Psychology: General*. 122, 2 (1993), 166–183.

[54] Tausczik, Y.R. and Pennebaker, J.W. 2010. The Psychological Meaning of Words: LIWC and Computerized Text Analysis Methods. *Journal of Language and Social Psychology*. 29, 1 (Mar. 2010), 24–54.

[55] Wang, H., Zhang, H. and Yu, D. 2023. On the Dimensionality of Sentence Embeddings. arXiv:2310.15285.

[56] Wei, J. et al. 2022. Emergent Abilities of Large Language Models. arXiv:2206.07682.

[57] Winne, P.H. 2010. Improving Measurements of Self-Regulated Learning. *Educational Psychologist*. 45, 4 (Oct. 2010), 267–276. DOI:https://doi.org/10.1080/00461520.2010.517150.

[58] Winne, P.H. 2017. Learning Analytics for Self-Regulated Learning. *Handbook of Learning Analytics*. (2017), 531–566.

[59] Winne, P.H. 2013. The Potentials of Educational Data Mining for Researching Metacognition, Motivation and Self-Regulated Learning. *Journal of Educational Data Mining*. 5, 1 (2013), 1–8.

[60] Winne, P.H. and Hadwin, A.F. 1998. Studying as Self-Regulated Learning. *Metacognition in Educational Theory and Practice*. (1998), 277–304.

[61] Yoo, Y. 2024. Automated Think-Aloud Protocol for Identifying Students with Reading Comprehension Impairment Using Sentence Embedding. *Applied Sciences*. 14, 2 (Jan. 2024), 858. DOI:https://doi.org/10.3390/app14020858.

[62] Zambrano, A.F., Liu, X., Barany, A., Baker, R.S., Kim, J. and Nasiar, N. 2023. From nCoder to ChatGPT: From Automated Coding to Refining Human Coding. *In International conference on quantitative ethnography* (2023), 470–485.

[63] Zhang, J., Andres, J.M.A.L., Hutt, S., Baker, R.S., Ocumpaugh, J., Nasiar, N., Mills, C., Brooks, J., Sethuaman, S. and Young, T. 2022. Using Machine Learning to Detect SMART Model Cognitive Operations in Mathematical Problem-Solving Process. *Journal of Educational Data Mining*. 14, 3 (2022), 76–108.

[64] Zhang, J., Baker, R.S., Andres, J.M.A., Hutt, S. and Sethuraman, S. 2023. Automated Multi-Dimensional Analysis of Peer Feedback in Middle School Mathematics. *16th International Conference on Computer-Supported Collaborative Learning (CSCL)* (Oct. 2023), 221–224.

[65] Zhang, J., Pang, S., Andres, J.M.A.L., Baker, R.S., Cloude, E.B., Nguyen, H. and McLaren, B.M. 2023. Leveraging Natural Language Processing to Detect Gaming the System in Open-ended Questions in a Math Digital Learning Game. *Presented at the 33rd Annual Meeting of the Society for Text and Discourse* (Oslo, Norway, 2023).

[66] Zheng, L. 2016. The effectiveness of self-regulated learning scaffolds on academic performance in computer-based learning environments: a meta-analysis. *Asia Pacific Education Review*. 17, 2 (Jun. 2016), 187–202. DOI:https://doi.org/10.1007/s12564-016-9426-9.

[67] Zimmerman, B.J. 2000. Attaining Self-Regulation: A Social Cognitive Perspective. *Handbook of Self-Regulation*. 13–39.

[68] Zimmerman, B.J. 1990. Self-Regulated Learning and Academic Achievement: An Overview. *Educational Psychologist*. 25, 1 (1990), 3–17.

[69] Zimmerman, B.J. and Schunk, D.H. 2011. Self-Regulated Learning and Performance. *Handbook of Self-Regulation of Learning and Performance*. 1–12.