Assessing Learning Behavior and Cognitive Bias from Web Logs

Rashmi Jayathirtha Rao[1], Christopher Stewart[1], Arnulfo Perez[2], and Siva Meenakshi Renganathan[1]

[1]Dept. of Computer Science and Engineering , The Ohio State University
[2]Dept. of Teaching and Learning, The Ohio State University

***Abstract*—This research to practice, work in progress paper presents the analysis strategy used to assess the learning behavior using logs on an e-learning platform. Students who can link algebraic functions to their corresponding graphs perform well in STEM courses. Early algebra curricula teaches these concepts in tandem. However, it is challenging to assess whether students are linking the concepts. Video analyses, interviews and other traditional methods that aim to quantify how students link the concepts taught in school require precious classroom and teacher time. We use web logs to infer learning. Web logs are widely available and amenable to data science. Our approach partitions the web interface into components related to data and graph concepts. We collect click and mouse movement data as users interact with these components. We used statistical and data mining techniques to model their learning behavior. We built our models to assess learning behavior for a workshop presented in Summer 2016. Students in the workshop were middle-school math teachers planning to use this curriculum in their own classrooms. We used our models to assess participation levels, a prerequisite indicator for learning. Our models aligned with ground-truth traditional methods for 17 of 18 students. The results of the models with respect to the two types of components of the web portal have been used to infer possible data or graph oriented cognitive bias.**

Keywords*—*Mathematical Modeling, Computational Thinking, Data Science, web log analytics

I. Introduction

Computational thinking is a problem solving approach wherein problems and solutions are formulated and represented in a fashion amenable to information processing agents [1]. It is imperative to develop computational thinking skills among science, technology, engineering and mathematics (STEM) educators and students to sustain scientific disciplines that increasingly depend on computational methods [2]. Linear algebra plays a pivotal role in STEM disciplines as it usually follows a calculus sequence that may be predominantly computational in focus. Students with skills in computational thinking often excel in algebra and STEM disciplines [3]. Hence it is essential to understand the learning behavior that students exhibit when introduced to computational thinking. By understanding learning behavior, educational researchers can investigate pedagogical approaches to inculcate computational thinking in the curriculum.

Recent research by the Dept. of Teaching and Learning at The Ohio State University [4] has studied computational thinking in the teachers and students in smart classroom environments. Henceforth, we refer to teacher and students in this study as participants. The participants of research were introduced to science experiments involving linear algebra. The participants entered observations from these experiments on a smart classroom platform developed on top of the Moodle platform. Moodle was used to manage the user login and provide access to the appropriate view of the web application which we address as the web portal in this paper. Once the observed data was entered, the results of the experiments were visualized graphically. For example, one of the experiments depicted Ohm's law [5]. In this experiment, the participants connected a resistor to a voltage source and measured current values(mA) for different amounts of voltage. The participants then entered the input voltage and observed current. They then visualized this observed data in the form of a graph. The graph, in this experiment, was a line created by linear regression of the data points entered by the user. The slope represented the resistance value. The users can perform several other functions on the portal like share their tables, import the tables shared by group mates, move sliders to change the angle of the regression line and even update the length of the axes on the graph. The participants of research are video graphed while performing the experiments. This allowed researchers to analyze each persons learning behavior by visually observing how they interacted with the smart classroom platform. Participants also took surveys and quizzes which tested their computational thinking levels. This analysis and interpretation reflects the traditional way of inferring computational thinking level. The participation of the teachers in the entire research experiment directly correlated with their computational thinking levels and hence we refer to the motivation towards computational thinking as the participation level.

The traditional way of analyzing the participation levels requires observing the videos and identifying the learning behavior of each student during the research experiment. This process is highly time consuming. Recent research involving usage of a web application for project management found that higher activity on the web portal correlated to higher motivation levels toward the project and correspondingly higher grades. [6]. The research depicts a relationship between a successful project outcome and the students' use of the web application. They report significant positive relationships between the students' performance with respect to the project outcomes and their interactions with that website. It is implicitly inferred that higher motivation levels towards the project will result in better project outcomes and this is reflected in the use of the website resources.

In our research, we used logs collected from the smart classroom web portal to predict participation levels. Such

978-1-5386-1174-6/18/$31.00 ©2018 IEEE

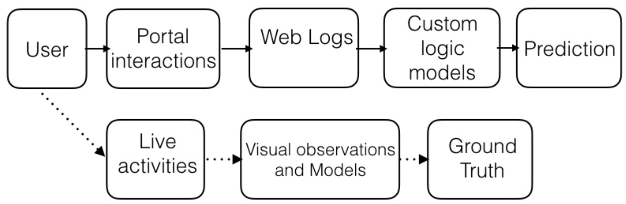

Fig. 1: Block Diagram of the assessment methodology. The dotted lines indicate the traditional approach.

automated analysis substantially reduces cost and delay to assess student learning[1]. A block diagram of the assessment methodology can be found in Figure 1.

To demonstrate the value of our approach, we used web log data to understand if a participant exhibited a learning bias towards either the data or graphical representation. For example, bias towards graphical representation indicates either higher interest or a struggle to understand the graphical concept. Research indicates that eye movement correlates with mouse movement during the use of a web page [7]. To capture the attention and learning behavior of the students on the portal, we use the mouse movement logs that are collected from the portal. We analyzed how the participants interacted with the smart classroom portal and specifically tried to understand how they linked the algebraic concepts of data and graph. Our work, thus far, is preliminary. Thus we do not draw conclusions about how participants learned. Instead, our contribution is a methodology to use web logs to assess learning behaviors. Key research questions addressed are:

- Which methods can correlate web log features to learning behavior?
- Are there data analysis methods that yield predictive models for learning behavior?
- Can smaller, early samples of web log data provide promising results for inferring learning behaviors in time for pedagogical interventions?

The remainder of the paper is as follows. Section II provides a cursory overview of related work. Section III describes our approach to engineer features from a real data set, outlining methods can be used in future studies. Section IV presents early results. Section V concludes.

II. Related work

Research by Werner et al. analyzed student-web interaction to measure the computational thinking skills in middle school students using the web log data [8]. The students took a battery of assessments with a gaming application and their actions were captured through web log data and recordings on the gaming application. Human graded rubric scores were used to assess each student's performance and it was found that assessment scores correlated with students' interest in taking a computer science class, confidence with computers and attitude toward computing. The key features of this dataset namely the human graded rubric scores and the collected web log data has been used to predict the high and low performing students [9]. The vocabulary of the code written by the students is used in modeling and predicting the assessment scores. In contrast, our work collected rich web log data, including mouse clicks and time stamps. Our challenge is linking content unaware log files to learning behavior. We extract features from our web logs that characterize complicated learning behavior and label the web log events with their intended learning effect. We then build models to classify students as highly or less motivated towards computational thinking. This further allows us to infer cognitive bias of learners that prefer graphs or data representations of the curriculum. MathWorks also uses log data to improve educational outcomes [10], but these efforts focus on improving student coding skills.

Web logs require fewer human resources than traditional techniques, however collection takes time. Our work makes early efforts in studying the impact of software-driven approximation by directly testing models on smaller datasets. Prior work in this space [11], [12], [13] measures the effect of approximation via memoization and focuses optimizing computation.

III. Feature Engineering and modeling

Feature engineering is the process of using domain knowledge of the data to create features[14]. The raw relational web log data has the information about set of actions performed by the users at different instants of time. But features need to be extracted from this data to represent learning behavior with respect to each student. Research suggests that key indicators of the learning behavior in web based learning are [6]:

- Web Resources: Which resources on the web were accessed
- Web access rate: How frequently the web page resources were accessed.
- Time spent with the resources
- Usage pattern of the resources

We define a metric called Score to quantify and depict the overall participation level and a metric called Sub score to depict participation level with respect to individual type of behavior for the comparative assessment of the students. We perform the learning behavior modeling on data from 18 students in the summer workshop. For additional details on our methodology, see [15].

A. *Web Resources*

Web resource is a facility provided in a web application[6] The main resources on our portal are mainly components used to enter, assemble data and the components used to visualize the data graphically. Hence we've divided each of the web components into either data or graph component as shown in Figure 2 We abbreviate data components as DC and graph components as GC.

B. *Web resource accesses*

This aspect considers how many times the web resources were accessed. Consider a pattern of access on the portal like

[1]IRB approved aspects of this study related to personally identifiable data

Label	Component Name	Associated Behaviour
DC1	My Tables - Table Name	Data
DC2	My Tables - Share	Data
DC3	My Tables - Delete	Data
DC4	Tables shared with me - Table name	Data
DC5	Tables Shared with me - Refresh	Data
DC6	Input Table	Data
DC7	Save Table	Data
GC1	Graph Button	Graphical
GC2	Graph - SVG	Graphical
GC3	Slider - Primary Table	Graphical
GC4	Update Axes	Graphical
GC5	Slider- Secondary Table	Graphical

Fig. 2: Web components and their behavior category

: DC1->GC1->DC2->GC2->DC3 and so on. This pattern shows that the student is equally accessing both the data and graphical portions of the application. To characterize this behavior, we developed a model that captures the number of accesses from data to the graph and the graph to the data oriented components on the page. The graph and the data sub scores derived is as shown below. Consider a network with node as the component and the directed edges as the accesses direction with weight equal to the total number of access from one component to the other. D_1 and G_1 denote the data and graph sub scores respectively

$$D_1 = \sum_{i \in DC, j \in GC} w_{ij}$$

$$G_1 = \sum_{i \in DC, j \in GC} w_{ji}$$

C. *Time spent with the resources*

Time spent by the students with different resources is another indicator of the point of attention on an e-learning application. Consider a pattern of access such as (DC6, 0)->(GC2,10)->(DC6 ,60)->(DC1, 65)->(DC6 ,70)->(GC3, 75)->(DC6 , 90) This representation shows the component accessed and the time stamp or the time at which it was accessed. So, it can be observed that a student might interact more with the same type of component but spend more time on the other type.

To characterize this behavior, the graph and the data sub scores derived is as shown below D_2 and G_2.

$$D_2 = \frac{\sum_{j \in DC} I_j T_j}{\sum_{i \in DC, GC} T_i}.$$

$$G_2 = \frac{\sum_{j \in GC} I_j T_j}{\sum_{i \in DC, GC} T_i}.$$

Where D_2 and G_2 are the second set of sub scores. Each sub score represents the total number of accesses between the same type of components scaled by the fraction of the time spent in that type of component.

D. *Pattern of access of the resources*

The previous sub scores considered the aspects of interaction counts within and in between resources, but there's a need to consider the actual order in which the resources were accessed and the number of data and graph components in them. To characterize this behavior, we devise the third model as the number of Data/Graph ($|V(DC)|$/$|V(GC)|$) components in the most frequently accessed patterns of length 6.

$$D_3 = \sum_{seq \in freqSequences} |V(DC)|_{seq}.$$

$$G_3 = \sum_{seq \in freqSequences} |V(GC)|_{seq}.$$

D_3 and G_3 denote the third data sub score and graph sub score. The pattern length was chosen empirically. We ran the Generalized sequential pattern mining algorithm on the access patterns of all the 18 students. We found significant number of frequently accessed patterns of length 6 for every student.

E. *Classification*

We considered each of the data and graph related features or sub scores of all the students that were extracted in the models to observe the distribution of the scores with respect to the classification. We observed that they are not linearly separable while a combination of 2 features and 3 features together were more linearly separable. Based on the separability observation, we came up with a model to combine the data and graph related features.

$$(GS, DS) = \left(\hat{G}, \hat{D}\right) \tag{1}$$

$$\hat{G} = f\left(G_1, G_2, G_3\right) \tag{2}$$

$$\hat{D} = f\left(D_1, D_2, D_3\right) \tag{3}$$

GS and DS are Graph and Data Scores respectively and G_1, G_2, G_3 are the graph scores obtained from model 1, 2 and so on. Similarly, D_1, D_2, D_3 are the data scores obtained from different models. More models could be devised to characterize further learning behavior and can easily be plugged into the above equation. This is left for future research. To combine the sub scores, we performed the function of dimensionality reduction technique called principle component analysis and considering the primary component that captures the maximum variance in the data. The graph and data score after performing PCA(Model1, Model2 , Model3) is as shown in Figure 3. The orange points(circular shape) denote the highly participating students and the green points(triangular shape) denote the less participating students based on the classification through visual analysis of classroom videos. A simple linear hyperplane of graph score+data score>=-0.8 classifies that student as "high" and graph score+data score<-0.8 classifies the student as "less" participating. By this combined model we're able to classify 17 out of 18 users accurately.

F. *Cognitive Bias*

We infer cognitive bias by observing the high value of graph or data score that has contributed to the student being classified as "high" or "less". From the Figure 3 we can observe that the point on the top left has been classified as high. But this can be inferred as due to high graph score in spite of low data score. We define a cut off for data and graph scores as -0.4. If a student is classified as high but has a data or graph

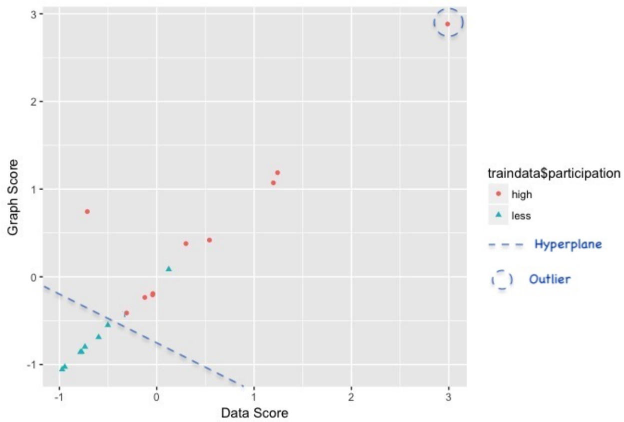

Fig. 3: Graph Score vs Data Score

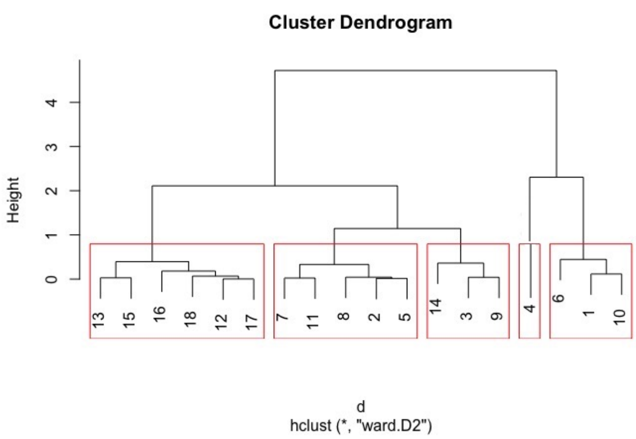

Fig. 4: Dendogram

score less than -0.4, its possible that he/she has a cognitive bias towards graph or data orientated learning respectively.

G. Outlier Detection

Its a hard task to group people of similar behavior and detect any outliers. As shown in Figure 3, the student on the top right is an outlier. To perform the outlier detection, we use the un-supervised method of K means clustering [16]. To get an idea of number of clusters we perform a hierarchical clustering and infer that K=5 for a cut of 0.9. This can be used to determine the outliers as shown in Figure 4. We picked the cluster with minimum number participants as the outliers. Hence we were able to identify student number 4 as the outlier.

IV. Classroom dataset results

We further applied the same models to relatively classify students in actual classrooms and made interesting observations. The applicability of the models built on a smaller data set onto a larger data set is supported by observing the distribution of the scores in both workshop and classroom data. We found that the mean and standard deviation of the data scores and graph scores of the students in workshop differed from those of the classroom scores by less than 4%. This analysis is useful to classify classrooms according to the computational thinking abilities of students. Further, it can identify students with cognitive bias for intervention. We observed that most of the classrooms have many highly participating students. The results were cross verified by referring to the field notes collected by the researchers. These notes are the notable observations in the behaviors or words expressed by the students in the classrooms while performing the science experiments. We compared the field notes of classes with students with lower participation scores and to those where most students had high participation scores. We found that the behaviors of the students correlate with our results. The field notes from classrooms with very high number number of highly participating students(based on the scores) indicated that many students in those classrooms were positively motivated towards the scientific experiments and computational thinking. On the other hand, the field notes from classrooms with very less number number of highly participating students indicated that many students didn't show attention or interest towards the experiments or computational thinking. Out of the 365 students considered, we identified 77 students with possible cognitive bias towards either graph or data. The hierarchical clustering mechanism produced 8 clusters with a minimum size 12, i.e, the students could be grouped into 8 clusters and 12 students could be identified as outliers. The web log data is large and keeps growing with more classrooms being introduced to the experiments and there's a need to compute the results using a smaller subset of the original data. We applied the models on smaller log datasets to analyze the effect of approximate computing. Approximate computing results in faster system performance. We considered different sizes of the classroom dataset(10,20...90% of the data) and performed in-order and out of order sampling. Error is calculated as:(Number of mis-predictions with sampled data when compared with predictions from complete data)/total number of predictions. It was observed that on an average, change in error is less than 0.5% when 70 % of the total data is considered for analysis. Hence we infer that 70% of the data sufficient to get results with minimum error and maximum performance gain. This threshold could be applied in real time to obtain results once 70% of the total data is available. Further research on approximation is left for future work.

V. Conclusion

We used web log data from a smart classroom portal to infer learning behaviors for 18 subjects in a study. We constructed several models from data mouse clicks and movement, including: resource access models, time spent per resource and access patterns. These models were able to accurately classify ground truth computational thinking labels for 17 out of the 18 students in the study. These models deserve further exploration. We also applied outlier detection methods to infer cognitive bias. The models can be calibrated on partial data and identified students that could benefit from pedagogical interventions. Such online sampling and approximation also deserves further research.

References

[1] J. Wing, "A definition of computational thinking from jeannette wing," 2016.

[2] S. I. Swaid, "Bringing computational thinking to stem education," in *6th International Conference on Applied Human Factors and Ergonomics and the Affiliated Conferences*, 2015.

[3] S. Bagley and J. M. Rabin, "Students' use of computational thinking in linear algebra," November 2015.

[4] Arnulfo Perez, Kathy Malone, Siva Meenakshi Renganathan,Kimberly Groshong , "Computer modeling and programming in algebra," *8th International Conference on Computer Supported Education*, 2010.

[5] Wikipedia, "Ohm's law," https://en.wikipedia.org/wiki/Ohm%27s_law, 2018 (accessed April 16, 2018).

[6] J. Sheard, J. Ceddia, J. Hurst, and J. Tuovinen, "Inferring student learning behaviour from website interactions: A usage analysis," *Education and Information Technologies*, 2003. [Online]. Available: https://doi.org/10.1023/A:1026360026073

[7] V. Navalpakkam, L. Jentzsch, R. Sayres, S. Ravi, A. Ahmed, and A. Smola, "Measurement and modeling of eye-mouse behavior in the presence of nonlinear page layouts," *Proceedings of the 22nd international conference on World Wide Web*, May 2013.

[8] L. L. Werner, J. Denner, S. Campe, and D. C. Kawamoto, "The fairy performance assessment: measuring computational thinking in middle school," *Proceedings of the 43rd ACM Technical Symposium on Computer Science Education*, 2012.

[9] N. Diana, M. Eagle, J. Stamper, S. Grover, M. Bienkowski, and S. Basu, "An instructor dashboard for real-time analytics in interactive programming assignments," *Learning Analytics and Knowledge*, 2017.

[10] C. Stewart, M. Leventi, and K. Shen, "Empirical examination of a collaborative web application," in *IEEE International Symposium on Workload Characterization*, 2008.

[11] J. Kelley, C. Stewart, N. Morris, D. Tiwari, Y. He, and S. Elnikety, "Measuring and managing answer quality for online data-intensive services," in *ICAC*, 2015.

[12] J. Kelley and C. Stewart and N. Morris and D. Tiwari and Y. He and S. Elnikety, "Obtaining and managing answer quality for online data-intensive services," in *ACM Transactions on Modeling and Performance Evaluation of Computing Systems*, 2017.

[13] Z. Xu, N. Deng, C. Stewart, and X. Wang, "Cadre: Carbon-aware data replication for geo-diverse services," in *IEEE International Conference on Autonomic Computing*, 2015.

[14] Wikipedia, "Feature engineering," https://en.wikipedia.org/wiki/Feature_engineering, 2018 (accessed April 16, 2018).

[15] R. Rao, "Modeling learning behaviour and cognitive bias from web logs," Master's thesis, The Ohio State University, http://library.ohio-state.edu/record=b8245815~S7, 2017.

[16] C. Stewart, K. Shen, A. Iyengar, and J. Yin, "Entomomodel: Understanding and avoiding performance anomaly manifestations," in *IEEE Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems*, 2010.