AC 2012-4663: DOES NEATNESS COUNT? WHAT THE ORGANIZATION OF STUDENT WORK SAYS ABOUT UNDERSTANDING

Mr. Timothy S. Van Arsdale, University of California, Riverside

Timothy Van Arsdale earned his B.S. in engineering from Walla Walla University in 2010. He is currently a Ph.D. student in mechanical wngineering at the University of California, Riverside.

Dr. Thomas Stahovich, University of California, Riverside

Thomas Stahovich received a B.S. in mechanical engineering from the University of California, Berkeley, in 1988. He received a M.S. and Ph.D. in mechanical engineering from the Massachusetts Institute of Technology in 1990 and 1995, respectively. He is currently Chair and professor in the Mechanical Engineering Department at the University of California, Riverside.

Does Neatness Count? What the Organization of Student Work Says About Understanding

Abstract

Students have long been taught that neatness counts. But does it? In this project, we seek to understand how the organization of a student's solution to a problem relates to the correctness of the work. Understanding this relationship will enable us to create software to provide early warnings to students who may be struggling in a course. In this study, students in an undergraduate statics course completed all of their work, including homework, quizzes, and exams, using Livescribe^{TM} Smartpens. These devices record the solutions as time-stamped pen strokes, enabling us to see not only the final ink on the page, but also the order in which it was written. Using this unique database of student work, we examine how the history of the solution construction process correlates with the correctness of the work. We characterize solution histories with a number of quantitative features describing the temporal and spatial organization of the work. For example, there are features that describe the order in which various problem solving activities, such as the construction of free body diagrams and equilibrium equations, are performed, and the amount of time spent on each activity. The spatial organization of the work is characterized by the extent to which a student revisits earlier parts of a solution to revise their work. Regression models have demonstrated that, on average, about 40% of the variance in student performance could be explained by our features. This is a surprising result in that the features consider only the process of recording the solution history and do not actually consider the semantics of the writing.

1 Introduction

Students have long been taught that neatness counts. But does it? In this project, we seek to understand how the organization of a student's solution to a problem relates to the quality of the solution. More precisely, we seek to understand how the history of the solution construction process correlates with the correctness of the work. Understanding this relationship will enable us to create software to provide early warnings to students who may be struggling in a course.

We have conducted a study in which students in an undergraduate statics course completed all of their work, including homework, quizzes, and exams, using Livescribe^{TM} Smartpens. These devices record the solutions as time-stamped pen strokes, enabling us to see not only the final ink on the page, but also the order in which it was written. While previous studies have used video cameras to record problem-solving activities, the analysis of such data is a difficult and time-consuming task that requires human judgment[1]. Capturing the work as time-stamped pen strokes enables a much more precise and efficient analysis of the work.

We seek to understand the relationship between how students construct their solutions and their performance on those problems. We refer to the sequence of problem-solving steps as a *solution history*. We characterize solution histories with a number of quantitative features describing the temporal and spatial organization of the work. For example, there are features that describe the order in which various problem solving activities, such as the construction of free body diagrams and equilibrium equations, are performed, and the amount of time spent on each activity.

Because Smartpens use ink, students cannot erase their errors and must cross them out. We characterize cross-outs by the delay between when the ink was written and when it was crossed out. The spatial organization of the work is characterized by the extent to which a student revisits earlier parts of a solution to revise the work. None of the features consider the actual correctness of the work. We then construct regression models to determine the extent to which these features correlate with correctness of the solution. In the examples we considered, on average about 40% of the variance in performance could be explained by these features. This is a surprising result in that the features consider only the process of recording the solution history and do not actually consider the semantics of the writing.

Section 2 places this work in the context of related work. Section 3 then describes the data we collected with the Livescribe^{TM} Smartpens. This is followed in Section 4 with a discussion of the features we used to characterize the temporal and spatial properties of a solution history. Section 5 presents the results of our regression analysis, which are then discussed in Section 6. Finally, Section 7 presents conclusions.

2 Related Work

Our work is a form of educational data mining, a research discipline that uses machine learning techniques, data mining techniques, and other similar techniques to examine education research issues. Romero and Ventura[2] provide a recent overview of work in this area. Much of this work relies on data collected in online environments such as web applications and intelligent tutoring systems. Our work is unique in that we use digital records of students' handwritten solutions, enabling us to study work habits in a more natural environment. The work of Oviatt *et al.*[3] suggests that natural work environments are critical to student performance. In their examinations of computer interfaces for completing geometry problems, they found that "as the interfaces departed more from familiar work practice..., students would experience greater cognitive load such that performance would deteriorate in speed, attentional focus, meta-cognitive control, correctness of problem solutions, and memory."

There have been several studies examining student work habits and performance in statics. For example, Steif and Dollár[4] examined usage patterns of a web-based statics tutoring system to determine the effects on learning. They found that learning gains increased with the number of tutorial elements completed. This study again relied on an online learning environment, while we consider ordinary handwritten work. In another study, Steif *et al.*[5] examined whether students can be induced to talk and think about the bodies in a statics problem, and if doing so can increase the student's performance. They used tablet PCs to record the students' spoken explanations and capture their handwritten solutions as time-stamped pen strokes. The study focused on the spoken explanations, with the record of written work left mostly unanalyzed.

Researchers have used video recordings to examine student problem solving. For example, Blanc[6] examined video recordings of student work in mathematics and analyzed the path that students used to solve an example problem. Although Blanc recorded more than 75 problem solutions, only two were analyzed in his paper. This speaks to the labor intensive nature of analyzing video records. Our pen stroke data is more amenable to automated computation.

Other researchers have used journaling to examine student work habits. For example, Orr *et al.*[7] examined students' journal responses about their study habits, including factors such as when and how they completed their homework, and if they took advantage of assistance programs. While the results proved interesting, journals capture students' perceptions of their work habits, rather than an objective characterization of them. Our work provides a nice complement to this work as we capture a detailed time-stamped record of a student's work over the duration of the course.

The ultimate goal of our work is to rapidly and inexpensively identify students who may be struggling in a course so that extra assistance can be provided. Other researchers have explored various mechanisms for providing rapid feedback. For example, Rasila *et al.*[8] explored the benefits of an online assessment tool for engineering mathematics. They found that automatic assessment was highly useful and improved the feedback provided to students. Chen *et al.*[9] used electronic conceptual quizzes during lectures within a Statics course to help guide the lecture content. They found that the rapid feedback produced a significant increase in student performance.

3 Data Set

We conducted a large-scale study in which over 120 students from an undergraduate mechanical engineering course in statics were given Livescribe^{TM} Smartpens. These devices serve the same function as a traditional ink pen, but additionally they digitize the pen strokes and store them as sequences of time-stamped coordinates. Students from this course were asked to complete all coursework using the pens. This included seven homework assignments with 44 problems in total; six quizzes with one problem each; and three exams with a total of 13 problems. The resulting digital database contains over four million pen strokes.

We restrict our present analysis to problems from the final exam, as data from quizzes and exams is more reliable than that from homework. It is possible that some students solved their homework on scratch paper, and then copied their work with the digital pen for final submission. This could not happen for quizzes and exams.

4 Characterizing Solution Histories

To examine the correlation between the properties of the solution histories and the correctness of the work, we first represent those properties quantitatively. We characterize a solution history in terms of both the temporal and spatial distribution of the work. The sections that follow describe the features we use for this purpose.

4.1 Solution History: Temporal Features

In characterizing the temporal distribution of the work in a solution history, we distinguish between six solution activities: drawing free body diagrams, constructing and solving equilibrium equations, making organizational marks, performing geometric computations, crossing out work, and working on other problems. Organizational marks are arrows, notes, lines, or other symbols students use to organize their solution but which are not part of the solution. Geometric computations include diagrams and equations used to determine geometric quantities, such as angles. The last of the six solution activities is when a student interrupts their effort on the problem at hand to work on another problem. In the current work, we manually label each pen stroke in a solution according to the solution activity it represents.

To represent the sequence of solution activities, we divide the problem solution into n, equal-time intervals. Each interval is labeled according to the solution activity that occurs most frequently during that interval, which is computed using the pen stroke labels. For example, if 70% of the drawing time in an interval was spent drawing free body diagram pen strokes, and the remaining time was spent drawing equation pen strokes, the interval as a whole would be characterized by the free body diagram activity. If no writing occurs during an interval, it is labeled as a break. In practice, we have found that using a value of 400 for n provides adequate detail to enable meaningful analysis of the solution. One advantage of this representation is that it abstracts away the total elapsed time, making it possible to directly compare the work of all students regardless of their total solution time.

If the student interrupts their work on a problem to work on other problems, we modify this representation slightly. If there are m such interruptions, we divide the work on the problem in question into $n - m$ equal intervals and compute their labels as before. Each of the m interruptions is then represented by an additional interval labeled as “other problem.” Figure 1 shows a portion of a typical activity sequence.

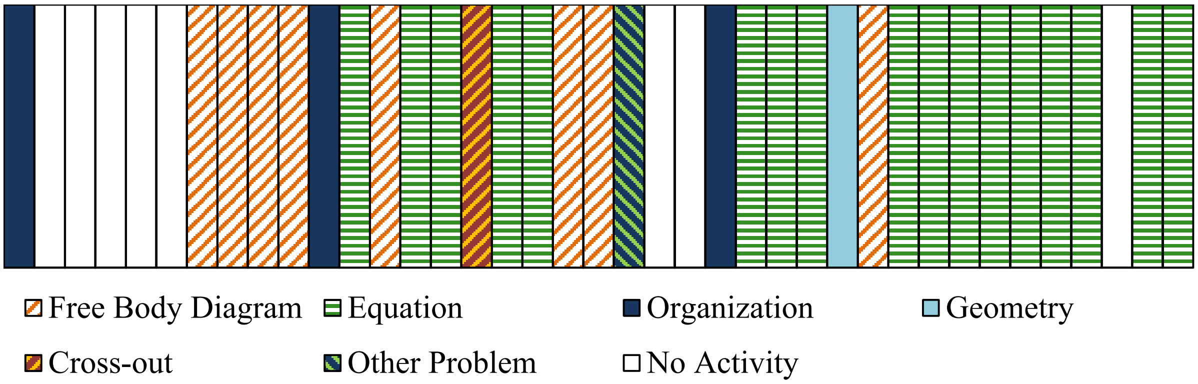

Figure 1: A portion of a typical discretized activity sequence.

The distribution of activities in the discretized solution history gives important insights into the student’s thought process. We have designed a set of six features to capture these insights. The first four features describe the amount of time spent on various activities. *FBD Effort* is the total number of activity intervals spent on free body diagrams, while *EQN Effort* is the number spent on equations. The *Break* feature is the number of intervals in which no work was done, while the *Other-Problem* feature is the number of times the student interrupted their work on the problem to work on other problems (this is the value “m” described above).

We have also created features that describe the sequencing of the activities. An expert might solve a problem by first constructing all of the free body diagrams, and then constructing all of the equations. This would result in a very simple activity distribution. A novice student who is struggling on a problem might repeatedly move from one activity to another in a much more complex pattern. We use information theory notions of entropy and complexity to capture these distinctions. We compute the *Entropy* of the sequence using the usual approach:

$$Entropy = \sum_{i} -(n_i/n)\, ln(n_i/n)$$

where n_i is the number of occurrences of a particular type of activity, n is the total number of activities (400), and the sum is taken over the six types of activities. (In this computation, we assume $ln(0) = 0$.) If the sequence contains, for example, only two types of activities, the entropy is relatively small. If, on the other hand, an equal amount of time is spent on each of the six types of activities, the entropy is maximal.

The Kolmogorov complexity[10] of a sequence is a measure of the minimum length required to describe it. To estimate this value, we first represent the sequence as a character string, assigning a unique letter to each of the six types of activities. We then use a standard data compression algorithm (the ZLIB[11] implementation of DEFLATE[12]) to compress the string. We define the *Complexity* of the sequence as the length of the compressed string. A random distribution of activities will result in a large value for this feature, while a distribution comprised of a few large blocks of activities will result in a small value.

4.2 Solution History: Spatial Features

The spatial organization of the solution on the page gives additional insights about the student's problem solving process. For example, a student who starts at the top of a page and progresses down it may understand the problem better than a student who revisits earlier work and revises it. We describe the spatial organization with four features that consider the progression of the work on the page and cross-outs.

We describe progression down the page in terms of deviation from a reference progression in which each stroke is drawn later than the ones above it. We use a sliding one-inch-tall window to construct this reference timeline (Figure 2). The window is initially placed at the top of the work. The reference time assigned to the location of the top of the window is computed as the time of the earliest stroke in the window. (The center point of a stroke's bounding box is used to determine if the stroke is in the window.) In the example in Figure 2, the time stamp of the pen stroke for the letter "P" in "problem" determines the time assigned to the top of the window. The window is then slid down the page a small distance. The reference time assigned to the new location of the top of the window is again that of the earliest stroke in the window, unless that is earlier than the time assigned to the previous window. In that case, the reference time is taken to be that of the previous window. The process is repeated for a total of 50 equally spaced positions of the window, resulting in 50 monotonically increasing reference time values, equally spaced between the top and bottom strokes of the solution.

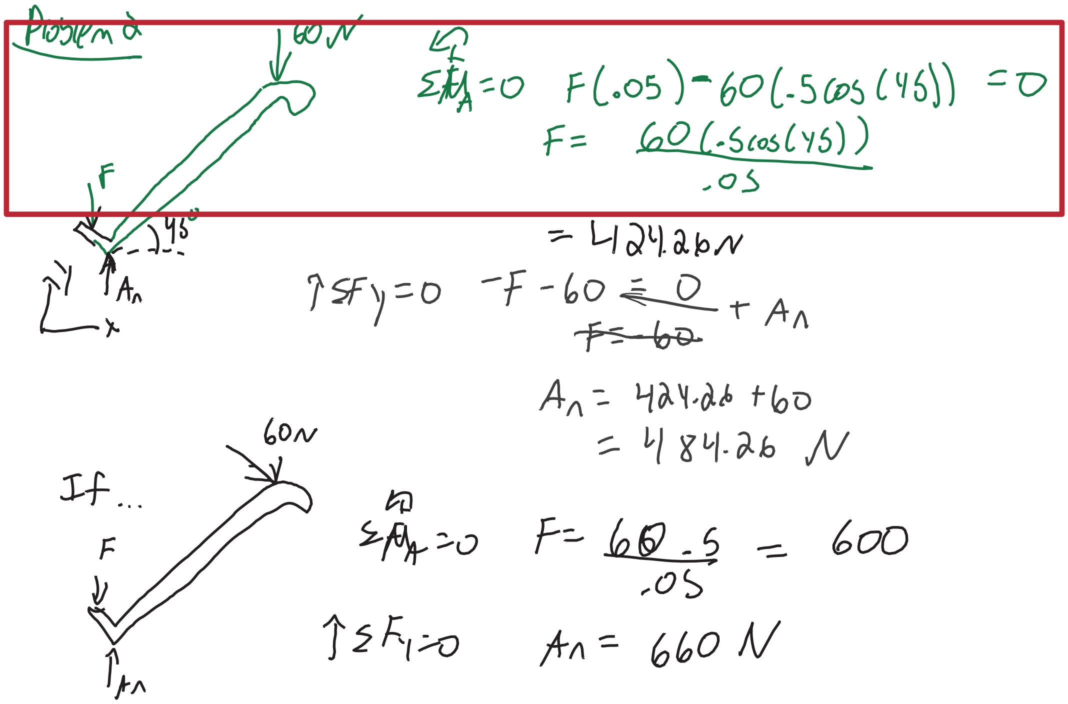

Figure 2: A sliding window (red box) is used to compute a reference timeline. The time stamp of the earliest stroke in the box is assigned to the location of the top of the box. Strokes inside the box are shown in green.

Once the reference timeline has been constructed, we compute the *Out-of-Order* feature, defined as the fraction of the pen strokes that were drawn out of order. To do this, we compute the reference time at each stroke's location on the page by linear interpolation of the reference timeline. If the time of a pen stroke differs from this reference by at least 30% of the total solution time, the stroke is considered to be out of order.

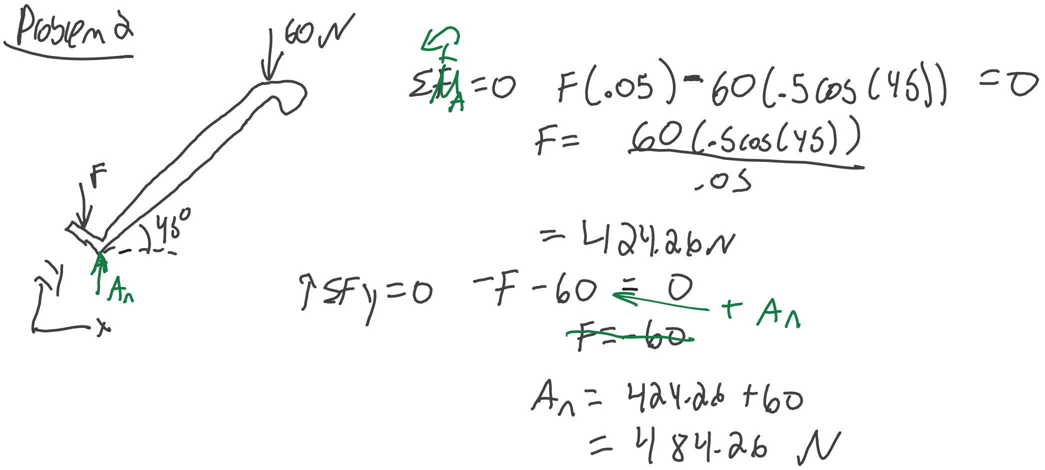

Figure 3: *Out-of-Order* work: In this hypothetical example, the student revised the free body diagram by adding an additional force after beginning the equilibrium equations. Out-of-order strokes are shown in green.

Cross-outs are a direct indication of revised work. We distinguish between two kinds of cross-outs, which we call "typo cross-outs" and "problem-solving cross-outs". The former are cases in which the student writes something and crosses it out within fifteen seconds, as if correcting a typographical error. The latter are cases in which there is a substantial delay (greater than fifteen seconds) between when the ink was written and when it was crossed out. These cases are more likely to be corrections of problem-solving errors. We characterize cross-outs with three features. The *Typo-Cross-Outs* and *PS-Cross-Outs* features are the numbers of typo and problem-solving cross-outs, respectively. The *Big-Cross-Outs* feature is the number of cross-outs that cover (delete) 10 or more pen strokes and thus represents a revision of a substantial amount work.

5 Results

We used the features described above to construct a variety of models to predict problem-solving performance. Specifically, we constructed three types of linear regression models: models considering the six temporal features, models considering the four spatial features, and models considering all 10 features. We used IBM® SPSS® Statistics version 20 to construct separate models for each of the 6 statics questions on the final exam. The grade assigned on each problem was used as the measure of problem-solving performance. Table 1 lists the coefficients of determination (R^2) for the models. The temporal features consistently produced stronger correlations than did the spatial features. Using the temporal features, R^2 ranged from 0.225 to 0.521, while for the spatial features it ranged from 0.081 to 0.285. On average, the temporal features explained 35.3% of the variance in performance, while the spatial features explained only 15.0%. Interestingly, the problems for which the temporal features were more strongly correlated with performance were also problems for which the spatial features produced the strongest correlation.

The temporal and spatial features together have greater predictive ability than either set alone. Thus, the two types of features do provide different information about student performance. Using the combined features, R^2 ranged from 0.280 to 0.578 (Table 1, Column 4). On average, the combined features explained 39.9% of the variance in performance.

Problem	**Six Temporal Features**	**Four Spatial Features**	**Ten Temporal and Spatial Features**	**Stepwise Feature Selection using Temporal and Spatial Features**
P1	0.468	0.146	0.479	0.408
P2	0.232	0.113	0.318	0.259
P3	0.225	0.081	0.280	0.245
P4	0.295	0.148	0.327	0.300
P5	0.374	0.129	0.412	0.330
P6	0.521	0.285	0.578	0.571
Average	0.353	0.150	0.399	0.352

Table 1: Coefficients of determination (R^2) for linear regression models using various feature sets to predict performance on final exam problems.

To determine which of the 10 features were the most important for predicting student performance, we performed a stepwise linear regression. The threshold for inclusion of a feature was a p-value less than 0.05 (based on a *t*-test), while the threshold for removal was a p-value greater than 0.10. The coefficients of determination for the six problems (Table 1, Column 5) ranged from 0.245 to 0.571. On average the stepwise models explained 35.2% of the variance in performance, slightly less than the 39.9% average for ordinary regression models that included all ten features.

Feature	**P1**	**P2**	**P3**	**P4**	**P5**	**P6**
Constant	0.087 (0.403)	0.790 (0.000)	-0.026 (0.927)	0.589 (0.000)	0.408 (0.000)	0.275 (0.000)
FBD Effort	--	--	--	--	--	--
EQU Effort	0.609 (0.000)	--	0.638 (0.000)	0.438 (0.000)	0.478 (0.000)	0.675 (0.000)
Break	--	--	0.372 (0.013)	--	--	--
Entropy	0.281 (0.000)	--	--	--	--	--
Complexity	--	--	--	--	--	--
Other-Problem	--	-0.387 (0.000)	--	--	--	--
Out-of-Order	--	-0.318 (0.000)	-0.255 (0.006)	-0.208 (0.023)	-0.219 (0.011)	-0.191 (0.020)
Typo-Cross-Outs	--	--	--	--	--	-0.182 (0.015)
PS-Cross-Outs	--	--	--	--	--	--
Big-Cross-Outs	--	0.178 (0.045)	--	--	--	--

Table 2: Regression models for the six exam problems: standardized coefficient, β, and p-value (in parentheses) from *t*-tests for features included in the stepwise linear models. Note: constant coefficients are not standardized.

Table 2 contains the standardized coefficients (β) with p-values for each of the six problems. The standardized coefficients are a measure of the strength of the influence of a feature on the performance. Examination of Table 2 reveals that only two of the ten features, *EQU Effort* and *Out-of-Order*, were consistently identified as significant in the stepwise models. The former was positively correlated with performance, indicating that students who spent more time on equations tended to perform better. The latter was negatively correlated, indicating that students whose work did not follow a monotonic progression down the page tended to do worse. It is possible that such students may have revisited earlier work to complete or correct it, or that their work may have simply been disorganized. The other eight features were not consistently selected in the stepwise models. The *Break*, *Entropy*, *Other-Problem*, *Typo-Cross-Outs*, and *Big-Cross-Outs* features were included in models only one time each. The *FBD Effort*, *Complexity,* and *PS-Cross-Outs* features were not selected in any of the models.

It is possible that some of the eight infrequently selected features are correlated with the two frequently selected features. For example, *EQU Effort* and *FBD Effort* are highly correlated. These are typically the two largest components of problem-solving activities and thus the values are approximately complements. To examine this issue, we constructed two additional sets of regression models, one using the two frequently selected features, and the other using the remaining eight features. The coefficients of determination for these models are listed in Table 3. For comparison, the table also includes the coefficients of determination for the models including all 10 features. The models constructed using *EQU Effort* and *Out-of-Order* performed similar to those using the other eight features. On average the models with the two features explained 30.4% of the variance in performance, while the models with the other eight explained 31.5%.

Problem	***EQU Effort Out-of-Order***	**Other 8 Features**	**All 10 features**
P1	0.344	0.373	0.479
P2	0.148	0.260	0.318
P3	0.195	0.110	0.280
P4	0.264	0.295	0.327
P5	0.330	0.318	0.412
P6	0.541	0.534	0.578
Average	0.304	0.315	0.399

Table 3: Coefficients of determination (R^2) for linear regression models using the two most consistently significant features and the remaining eight features. For comparison, the coefficients of determination for models including all 10 features are also included.

6 Discussion

Our results reveal that on average about 40% of the variance in students' performance on the final exam questions in our experiment can be explained solely by the temporal and spatial organization of the work. This is a surprising result in that none of the features we use actually consider the correctness of the work.

Most of the predictive power comes from just two features, the amount of time spent on equations (*EQU Effort*) and the fraction of the work that was written out of order (*Out-of-Order*). Performance increased with effort on equations and decreased with out-of-order work. Our analysis suggests that the other eight temporal and spatial features still have value, but may be correlated with these two features.

In our continued work, we plan to improve our features and develop additional ones to better characterize a solution history. For example, we anticipated that the complexity feature would be more useful than it was. It is possible that this feature may confuse some forms of highly organized work with disorganized work. For example, when a student alternates between drawing free body diagrams and writing the associated equilibrium equations, the complexity will be large, just as if the student worked in an unstructured fashion. A more effective complexity feature might consider both the temporal and spatial characteristics of the work. For example, alternating between free body diagrams and equations might still be considered to have low complexity if the work progresses smoothly down the page.

Our *Out-of-Order* feature performed well, and in fact was one of the two most powerful features for predicting performance. However, it too has an obvious limitation: work written in multiple columns is mischaracterized as disorganized. We plan to develop techniques to detect such work so that each successive column can be treated as a continuation of the previous one.

The *Typo-Cross-Outs* and *PS-Cross-Outs* features characterize corrections in terms of the time delay between when the ink is written and when it is crossed out. Short delays are assumed to represent typographical corrections while longer delays are assumed to represent corrections to errors that are more conceptual in nature. Accurately distinguishing between these two types of corrections may require consideration of the scope of the correction. For example, crossing out a single pen stroke after a long delay may still represent a typographical correction.

Our current analysis included only problems from the final exam. We plan to extend this analysis to homework, quiz, and midterm problems. Homework may pose new challenges as students may have started their homework on scratch paper before writing their final solutions.

In a related project, we have developed tools to automatically recognize the pen strokes in a handwritten statics solution and label them as free body diagram strokes, equation strokes, and cross-outs. In our experiments, this tool achieves an accuracy of about 93% at this task. This tool will enable fully automatic analysis of solution histories.

7 Conclusion

We have examined how the organization of a student's solution to a problem relates to the correctness of the work. In this study, students in an undergraduate statics course completed all of their work, including homework, quizzes, and exams, using digital pens that recorded the work as time-stamped pen strokes. We characterized the solution history of each problem with a number of quantitative features describing the temporal and spatial organization of the work. Regression models revealed that, on average, about 40% of the variance in student performance could be explained by these features. Most of the predictive power comes from just two features, the amount of time spent on equations (*EQU Effort*) and the fraction of the work that was written out of order (*Out-of-Order*). On average, models using just these two features explained 30.4% of the variance in performance. These encouraging results demonstrate the feasibility of creating an automated assessment system that can inexpensively identify students who may be struggling in a course and need extra support.

8 References

[1] Rogers Hall. Video Recording as Theory. In D. Lesh and A. Kelly (Eds.) *Handbook of Research Design in Mathematics and Science Education,* 647-664. Mahweh, NJ: Lawrence Erlbaum, 2000.

[2] Cristóbal Romero and Sebastián Ventura. Educational data mining: A Review of the State of the Art. *IEEE Transactions on Systems, Man, and Cybernetics – Part C: Applications and Reviews,* 40(6):601-618, 2010.

[3] Sharon Oviatt, Alex Arthur, and Julia Cohen. Quiet interfaces that help students think. In *Proceedings of the 19th annual ACM symposium on user interface software and technology (UIST '06)*, 191–200, New York, NY, USA, 2006.

[4] Paul Steif and Anna Dollár. Study of Usage Patterns and Learning Gains in a Web-based Interactive Static Course. *Journal of Engineering Education*, 94(4):321-333, 2009.

[5] Paul Steif, Jamie Lobue, Anne Fay, and Levent Kara. Improving Problem Solving Performance by Inducing Talk about Salient Problem Features. *Journal of Engineering Education*, 99(2):135-142, 2010.

[6] Paul Blanc. Solving a Non-routine Problem: What Helps, What Hinders? In *Proceedings of the British Society for Research into Learning Mathematics,* 19(2):1-6, 1999.

[7] Marisa Orr, Lisa Benson, Matthew Ohland, and Sherrill Biggers. Student Study Habits and Their Effectiveness in an Integrated Statics and Dynamics Class. In *Proceedings of the 2008 American Society for Engineering Education Annual Conference and Exposition*, 2008.

[8] Antti Rasila, Linda Havola, Helle Majander, and Pekka Alestalo. Automatic assessment in engineering mathematics: evaluation of the impact. In Myller, E. (ed.), *ReflekTori 2010 Symposium of Engineering Education*, 37-45. Aalto University School of Science and Technology, 2010.

[9] John Chen, Dexter Whittinghill, and Jennifer Kadlowec. Classes that click: Fast, rich feedback to enhance student learning and satisfaction. *Journal of Engineering Education*, 99(2):159-168, 2010.

[10] C. S. Wallace and D. L. Dowe. Minimum Message Length and Kolmogorov Complexity. *Computer Journal*, 42(4):270-283, 1999.

[11] P. Deutsch and J. L. Gailly. ZLIB Compressed Data Format Specification version 3.3. *RFC Editor*, 1996.

[12] P. Deutsch. DEFLATE Compressed Data Format Specification version 1.3. RFC 1951, Aladdin Enterprises, May 1996.