Methods for Observing the Behavior of Computer Users

R. David Sigsby
Grand Valley State University

Follow this and additional works at: http://scholarworks.gvsu.edu/mcnair

Recommended Citation
Available at: http://scholarworks.gvsu.edu/mcnair/vol4/iss1/6
ABSTRACT
A large amount of research has been done to improve Intelligent Tutoring Systems (I.T.S.). Though implementing the latest Artificial Intelligence (A.I.) techniques, I.T.S.’s fail to approach the adaptability of human teachers. New methods are needed for observing computer users, analyzing and interpreting their behavior, and effectively offering online instruction.

This project, called Mental, is aimed at finding new methods for identifying observable behavior and what can be learned from those observations. Time stamped keystrokes were collected from a small group of Introduction to Computer Programming students for various assignments over the course of one semester. Specific instances of behavior (e.g., run, arrow keys, deletes, etc.) were then graphed in order to look at the student’s behavior as he or she progressed through the semester, as well as to compare the students to each other. The goals of this study are (1) to find behavioral patterns in computer use, and (2) to set the groundwork for a method of observation and analysis that can be generalized for various applications.

Introduction
When computers were first used as tools, they were primarily the domain of large corporations, government agencies, and the academic elite. Within the past decade, however, computers have become an integral part of modern life for everyone. Computer related skills have become necessary in all job markets and affect many aspects of our personal lives. While computers are becoming more powerful and more prevalent, the ratio of qualified teachers to those needing to be taught has increased, leaving many who need these skills without the necessary resources to learn them. Computers have been used as instructional tools to fill in for this shortage of qualified teachers. The ideal computer based solution to this problem are systems that would observe, assess, and aid the learning process, i.e., “systems that are capable of providing the quality of support of trainers but are as cheap and accessible as manuals.” [Mallen] Research has been done on using computers as instructional aids, evolving into programs that are called intelligent tutoring systems (I.T.S.). [Smith]

Existing I.T.S. technology is overdue for advancement: “[t]o be able to gather information on how people think about problems and present solutions modeled on that.” [Nass] Systems such as GUIDON (1983), WEST (1982), LISP TUTOR (1985), and BUGGY (1978) [Smith] have been designed to gather information about how the user learns and adapt instruction to the user’s ability or learning style. However, these systems have not been able to aid the learning process as proficiently as a human teacher. An I.T.S. must also be developed from the ground up, translating into high development costs and making such systems inaccessible to most people. Due to these reasons, I.T.S. technology has not seen widespread implementation or use.
This project, called Mentor, is aimed at making I.T.S. technology more effective and more easily available. It is the goal to achieve this by developing a strategy for observing computer users and identifying what portions of computer use can be used as behavioral variables (e.g., frequency of delete keys). In other words, our goal is to observe some form of computer use, identify behavioral variables, identify thought processes associated with those variables, and verify the results by statistical analysis. These variables will then be used to determine the timing with which help is offered to the user. Though Mentor will first focus on computer science students being taught computer programming, it is expected that the results can be generalized to other areas of computer use.

Methods
Our sample consisted of four students of differing ability from two CS 162 (introduction to computer programming) labs at Grand Valley State University. The students' ability is rated based upon individual lab grades, overall grade in the class, and professor evaluations of the student's performance and programming skill. The students used the programming environment JAVA Workshop for the Windows 98 operating system. During each lab, students were assigned a project aimed at developing their programming and problem solving skills. Data collection occurred during the winter semester of 2000.

The method of observation involved the capture of time stamped keystrokes for all four students. Capturing the student's use of the mouse could also be done, but this presented too many technical difficulties to properly implement at this time. A commercial program [Birjukov] served as the means to capture the keystrokes and gave the additional benefit of capturing windows titles when a window was opened or activated. A custom program added a time stamp to each character as it was captured and placed them into a file. Once the raw data (containing time stamps for keystrokes typed and windows activated by the student) had been collected, behavioral variables were identified and extracted for analysis. Preliminary reviewing of the data along with intuitive analysis of the tasks the students undertake allowed for identification of four behavioral variables: (1) runs, (2) right and left arrow keys, (3) up and down arrow keys, and (4) deletes and backspaces. It is believed that these variables gave insight into the students thought process, i.e., runs can show the student's breakdown of the assigned problem, deletes and backspaces show a student correcting mistakes, etc.

The final step in this portion of the project was analysis of the variables, which were plotted as a set of scatter plots with the x-axis representing time and the y-axis representing the variable type (see figure 1). Variables with related meanings were plotted with the same y value, i.e., up and down arrow keys are on the same line of the plot. A baseline of all captured data was plotted as well to insure data integrity. The plots allowed for a preliminary visual analysis of the variables to identify possible correlations. Statistical analysis of the behavioral variables is planned in order to confirm these preliminary findings.

Discussion
In order to choose variables from the raw data, we were led to make some preliminary hypotheses. The first of these is that running a program as it is being developed shows a significant break in the problem solving process. The second is that up and down arrow keys show that a student is scanning through their code to make additions or corrections. Next, we assume that right and left arrow keys signify a student has found a mistake and is focused on fixing the problem. Lastly, backspaces and deletes show a logical or syntactical mistake being corrected. The professor has ranked each student's ability based on careful initial observations, with student 1 the most competent and student 4 the least competent. These rankings will be verified by a more in depth review of each student, individual lab grades, and overall grades in the class.

The figures used are plots of the behavioral variables and were selected to show the trends discussed above. The x-axis represents time. Each line of the y-axis consists of a group of related variables, listed from the bottom to the top as follows: the baseline (all data recorded), deletes and backspaces, right and left arrow keys, up and down arrow keys, and runs.

Upon visual inspection of the plots there is meaning to the appearance of structure within the plotted variables. Figure 1 shows assignment 1 for students 1 and 2. The most visible difference is the number and spacing of the runs. The more competent student (student 1) begins to run the program after a large portion of code has been written, as evident by the large amount of data before the first run. There are a significant number of runs after this point spaced at regular intervals, with obvious editing occurring between each run. This seems to indicate that student 1 has a clear goal in mind and a process for achieving that goal. Student 2 shows runs clumped together in several groups with editing between these groups. In stark contrast, student 4 (not shown) never attempted to run their program at all. It is this degradation in the regularity and spacing of runs that lead to the conclusion that this variable gives significant insight into the students thought process and approach to solving the lab assignment.
A common trend visible, shown in figure 2 by the dotted circles, is the structure comprised of a run, followed by groups of up and down arrow keys, right and left arrow keys, and finally deletes and backspaces. While there is often overlapping of the timing between these groups of variables, there is a general order in which these groups are initiated. This structure occurs frequently for most students, with the exception of student 4 who has no connection between runs and writing or editing code.

Figure 3 shows project 3 for student 1. According to the lab professor, this is the most difficult assignment. It is interesting that this is the only assignment for which student 1 does not have evenly spaced runs. It is believed this represents a breakdown in the student's approach to solving the assignment. This shows a change in the student's behavior when compared with the other labs for the same student. We believe this signifies a breakdown in the student's ability to solve the problem.

In contrast, student 4 shows runs for only one assignment out of the three that were recorded. This is assignment 3, shown in figure 4, in which the student had minimal runs all at the beginning of the assignment. This suggests that student 4 has an idea of how to approach the other assignments but did not know how to approach the more difficult assignment.
Conclusions
The results are most clearly defined by students 1 and 4. Both students show a difference in behavior from the easier projects to the more difficult project. It is also evident that ability can be determined by the amount of structure to the data, as shown by the definite structure for the most competent student, student 1, and the lack of structure for the least competent student, student 4 (see figure 1). This structure can also be used to identify the student's breakdown of a problem. This is important since we are looking to identify appropriate times and ways to intervene in the learning process. While nothing has been done to directly implement the results into a finished product, an I.T.S. can use these principles of change in the structure of a student's behavior to decide when instruction is to be given. While these results are preliminary, it is reassuring to note that visible results can be seen in spite of the shortcomings of this project.

There were many technical problems encountered during the course of this project. Security issues associated with capturing keystrokes resulted in the use of laptops rather than lab computers for data collection. Only two laptops were available for use, which resulted in a small sample size. Another problem encountered was that data could not be collected for each student for every assignment due to a few minor technical problems and student absences. This would not have been as large of a concern with a larger sample size. Lastly, this study is directed at computer science applications, specifically computer programming. Some of the specific results apply directly to that field and are not able to be generalized. With this in mind, this project is to serve as a pilot study leading to a more in depth research. Further studies will generalize the methods to other applications.

Future Work
While it has been shown that results are visible, these results must be quantified in order to be implemented. For this reason, statistical analysis will be
performed on the existing data. Mean and standard deviations are currently the main focus of analysis, though other analyses will be considered as the need arises. This analysis completes the preliminary portion of this project. The experiment will then need to be improved upon and repeated. The first step towards improvement is to use a larger sample size and insure consistent data collection from all students for all labs. A more in depth study also expands the number of variables being extracted from keystrokes, as well as recording other types of input such as the mouse. Pressure sensitive keys, a camera, voice interaction, and even stress monitoring are more complex sources of input that could be examined. Observations also need to be made for other computer applications, i.e., word processing, database, and Internet use in order to generalize the methods to other uses of computers.

Another important phase of future work is to implement a test program that dynamically performs the observation and analysis. Such a program would make assumptions on the users ability that can then be verified by professor evaluations of the student. More importantly, such a test program would also incorporate a means of offering help to the user. Since previous research on I.T.S.'s have focused on the content of help being offered, it is the goal of this project to focus instead on effective timing of such help. This will be accomplished by incorporating a system of feedback from the user. This feedback can be used as a measure of how much the help interrupted the users normal thought process. It would be expected that the longer the program observed the user, the more accurate it will be in finding the appropriate times to interrupt. If successful, this test program can then become the model for implementing such a system into an I.T.S. in order to make them more efficient and more helpful.

References


