Raspberry Pi based learning center usage tracking system for optimal resource allocation

1st Aneet Dharmavaram Narendranath
Mechanical Engineering-Engineering Mechanics
Michigan Technological University
Houghton, MI, USA
dnaneet@mtu.edu

***Abstract*—The objective of this "work-in-progress" paper is to describe a Raspberry Pi based learning center management system that assists in deciding resource allocation for a peer-instruction driven engineering tutoring center. Data gathered from the usage of the engineering tutoring center allows for cost-effective staffing through recognition of the center's usage trends. The data collected may be structured per the center administrator's requirements and presented real-time on dashboards. The flexibility of this Raspberry Pi system allows for it to be developed into a full fledged learning management system with integrated analytics as required for a particular curriculum. This system is highly cost-effective as compared a commercial web-based system. This prototype has saved approximately 12 hours a month of time, for data retrieval and analysis as compared to a previously used commercial web-based system. At the moment, this Raspberry Pi data tracking system has been installed only in a single tutoring center on-campus. Deployment of this across a university campus would allow accumulated usage trends to be examined through data analysis operations such as seasonality and trend decomposition.**

***Index Terms*—Primary Topics: Computer-Based Learning and Courseware Technologies. Secondary Topics: Learning Analytics; Discipline Specific Issues: Data Science and/or Data Engineering; Discipline Specific Issues: Information Technology.**

I. Introduction

To measure the utilization of tutoring resources, in our peer-tutoring learning center (the engineering learning center/ELC) a Raspberry-Pi based learning center management system is initiated and installed. Preliminary data collected on the utilization of the learning center allows us to make data-driven staffing to provide targetted help for certain courses and certain times of week. The importance of peer driven instruction at our learning center and impact of education data-analytics are covered in the following paragraphs, in this introduction section. The current learning center management system and it's drawbacks that are corrected by our Raspberry-Pi based system are clarified in subsequent sections. A description of our device, and the home-grown, backend software and some preliminary results on learning center utilization quantification are discussed.

The benefit of peer driven instruction at learning centers in engineering departments, has been evaluated rigorously over the last few decades. Peer driven instruction or peer-tutoring is when students tutor each other [7]. It was shown that peer driven instruction at learning centers led to higher retention of minority students [1]–[3], led to positive influence on academic performance [4] through the enhancement of student learning process [5], [6] while being a cost-effective means of providing out-of-classroom instruction [8].

Our learning center relies on peer-tutoring but we are constrained by a budget (for tutor salaries and miscellaneous learning center expenses). It is important for us to not exceed our budget while providing peer-tutoring services for an ever growing population of students. To maintain or improve our level of service, it is necessary to collect data on the usage of our learning center and use this data to make targetted staffing of tutors.

Big data will play/is playing a big role in higher education through providing information that allows for timely reform in higher education through identification of avenues or processes that allow educators to improve their teaching leading to improvement in student learning. Data analytics in higher education may be one way to "penetrate the fog of uncertainty in higher education" [9] through creation of models, automated reporting [10]–[12]. In this work-in-progress (WIP) paper, we discuss an effort to collect "learning center usage" data.

Although, at this stage, the data we have collected and analyzed using our Raspberry-Pi based system may not be considered "big data" since we have not collected unstructured data of the order of gigabytes, the computer code developed to structure, visualize and analyse this data may be expanded for big data situations.

With our effort, we have been able to identify days and times of the week and courses that recorded most utilization of the learning center for. Such information was not readily available through the previous learning center management software, without significant time and effort exertion for every data retrieval. Such identification allows staffing to be made to provide targetted help in the future.

This Raspberry-Pi based system has multifold importance. Given the easily procurable and inexpensive nature of this single board computer and the open-source Debian Linux based Raspbian operating system, our system is significantly more modifiable and extensible, as described in this paper, as compared to the commercial software it has replaced.

II. Current system used in learning centers for data collection and management

978-1-5386-1174-6/18/$31.00

When a student visits our learning center, s/he uses the academic center management system from a web-browser window on a PC and selects from an on-screen list, the course or subject s/he needs help with, before proceeding to any available tutor. This process constitutes a "sign-in" and represents "usage of the learning center". Non-directory/private data is also collected by the commercial software and stored off-campus with the vendor of this software.

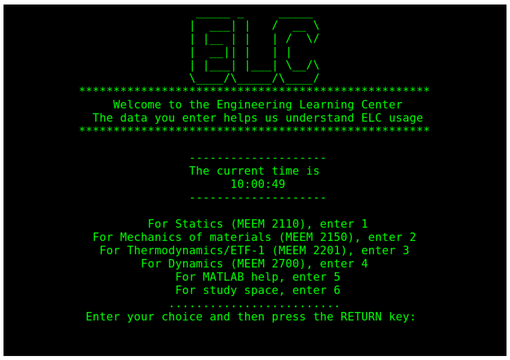

Fig. 1. Front end that user (student visiting the ELC) interacts with. This is wholly written as a BASH script and replaces the commercial software..

A. *Deficiencies of current system used in learning centers for usage data collection*

The current academic/learning center management system software being used across campus at our university to manage learning centers is expensive. It costs approximately $ 3700 for one learning center per year. This is approximately 15% of our yearly budget and represents 3 weeks of operation of the learning center with 12-16 tutors for 8 hours a day, 5 days a week. The data may only be collected or accessed through a web-browser and allows for compromising of sensitive student information, if proper log-out is not performed. Reporting is not automated and needs approximately half a dozen to a dozen mouse point-and-click operations, with each report generation consuming anywhere between 8-15 minutes per operation, depending on the familiarity of the user with the system. The system needs to be wholly manually operated without any possibility of scripting for automation and needed as many as 12 hours of interaction a month, if done on a daily basis. This is an inefficient use of time. The automation-through-scripting capability of the Raspberry-Pi device allows us to save on this time.

B. *Solution and improvements through the Raspberry-Pi based learning center management system*

To collect learning center usage data and to alleviate the need for regular human interaction to retrieve data and reports and to include automated, timed report generation and dashboards of learning center usage, we have created our own learning center management system and automation scripts deployed in our learning center, on a Raspberry-Pi. Moreover, non-directory/private information is not collected or stored, nullifying the risk of loss of sensitive data. This device, allows for a recognition of resource needs for different courses that the learning center caters to, high-frequency visitation times of day and their fulfillment in an optimized fashion.

The Raspberry-Pi computer, in today's day and age, requires little introduction. A web-search of scholarly documents on key phrases such as "impact of raspberry pi in education" reveals numerous highly cited scholarly articles in journals and conference proceedings. However, at the time of writing of this paper, no evidence of an academic/learning center management system deployed on a Raspberry Pi was uncovered. The author believes that our device and our automation scripts are an important step towards filling this gap. At the time of writing of this WIP paper, the following data is collected by our Raspberry-Pi based LMS system: (1) Student course preference, (2) The time (hour) and day of visit is collected. Eg: If a student visits between 9.00-10.00am on a Tuesday, and uses the terminal to log his/her visit, this is collected as "9, Tuesday".

Our automation scripts in Python and Mathematica process this data on the hour and generate usage heat maps (see fig 2 for example). Preliminary back-end code has also been developed in Octave to consolidate and generate plots and graphs on hours worked by tutors. This is not yet an active component. Once fully debugged, this would allow our Raspberry-Pi based initiative to also function as a "time-clock" software for coaches. Raspberry-Pi system logs are generated hourly and available to the administrator via the cloud-storage service Dropbox. This allows for two things:(1) provides notification that the Raspberry-Pi device is functional and running (2) Catching and rectifying any run-errors or operating system failures. Dropbox has been used to demonstrate the utility of cloud based storage. Using the rsync linux utility, this data may also be stored on designated on-campus network storage devices without issue. As stated previously, no sensitive information is collected or stored in dropbox.

III. Description of Device

In the following sections, the Raspberry-Pi (R-Pi) system, hardware peripherals and the flowchart for the collection of and operations on usage-data of the engineering learning center (ELC) are described.

A. *Hardware*

This effort uses the Raspberry-Pi 3.0 B model. This R-Pi is connected to the university network using a regular LAN cable. See table I for hardware list and costs.

B. *Operating system specifications*

The latest version (as of April 16, 2018) of the Raspbian operating system is installed on the Raspberry-Pi. The on-board version of Mathematica is used for plotting and visualizing data. Octave version 4.0 and Python version 2.7.9 are installed and bash version 4.3.30(1)-release (arm-unknown-linux-gnueabihf) is utilized for our command-line interface. Additionally the dropbox-uploader bash script is installed [13] allowing for automated back-up of data. Linux terminal

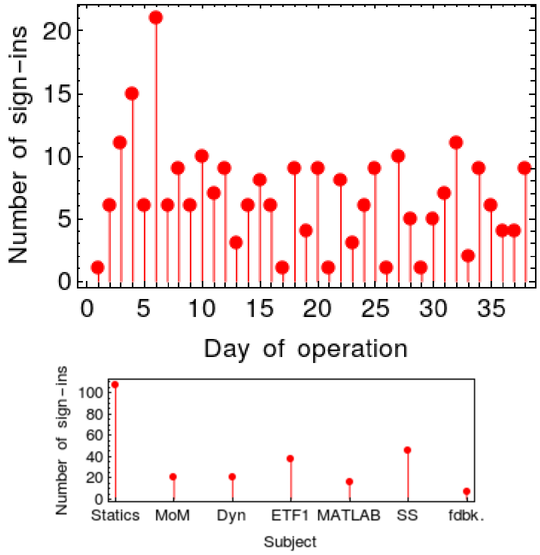

Fig. 2. Dashboard: plots that are generated hourly. X Axis: subject codes, Y Axis: number of sign-ins. At the moment, days without sign-ins and Sundays are not counted as "days of operation".

utilities, "rsync" and "sshpass" relay hourly data for web-publishing of plots.

C. Automation of data collection and processing

The program consists of a front-end (fig 1) that the user interacts with and a backend that performs plotting, back-up and publishing of usage data in HTML dashboards (fig 2). The operation of the front-end that students interact with, is described in flowchart figure 7 whilst that of the backend, that performs scheduled (through crontab) data analysis, visualization and back-up, is described in flowchart figure 8.

The front-end is written as a BASH script that runs perpetually. This script will be made available via a project github page for any other users to find and replicate for their purposes. The data collected by the front-end through user interaction is stored as comma separated value (CSV) files that are backed-up every 60 minutes to dropbox through the dropbox-uploader script [13]. The author is also experimenting with .JSON files as an alternative method to store data. This may be useful when the use of this device is expanded across the university campus, particularly if there needs to exist some hierarchical organization of usage data. At the moment, the author is not aware of what this hierarchical organization would be. Once the CSV data files are deposited in dropbox, a Mathematica script imports the data and plots it as shown in figure 2. These usage plots are then updated hourly on/synced to a web-server through rsync and an HTML displays the usage.

IV. Preliminary Results

Usage data of the ELC shows that Statics (engineering mechanics) is the prime driver of ELC usage with the number of students requiring statics help significantly outnumbering other courses and services (fig 2). If this trend continues, it would make sense to hire more tutors who are proficient in

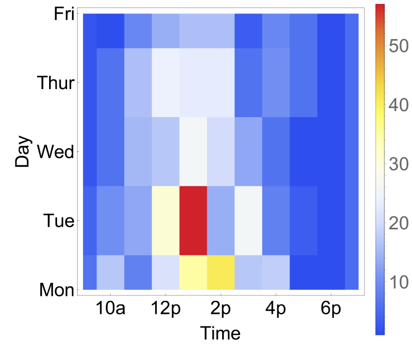

Fig. 3. Sample heat map showing significant ELC usage between 12-2pm on Tuesdays as compared to other days.

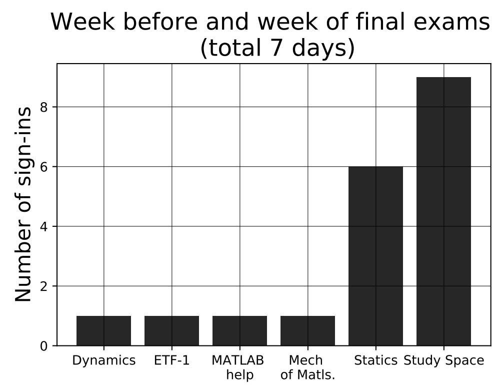

Fig. 4. Histogram that shows underwhelming usage of the ELC 7 days before exams.

statics. The ELC usage heat map in figure 3 shows that the ELC is most utilized between 12.00-2.00pm on Tuesdays. This could be correlated to the deadline for statics homeworks being Tuesday afternoons or Wednesdays. Assigning extra tutors to high frequency hours or on days before a statics homework is due, could prove useful to students who require help.

Interestingly, the week before and the week of final exams (fig 4), which had a total of 7 days of ELC functioning showed lower average use as compared to other weeks. Average number of sign-ins per day during days until a week before final exams was 6.7 but the sign-ins recorded during these two weeks before final exams was 2.7. This low usage rate was also validated by visiting the learning center and taking note of the number of students at different hours.

V. Notable Advantages of Raspberry Pi Based Learning Center Management System

The R-Pi based learning center management system allows for a large amount of flexibility and automation through BASH, Python, Mathematica and Octave scripts. The web-browser based, commercial system offers no such flexibility or scope of automation. Our GUI and analysis scripts will be distributed as open-source software through a github project page. This system may be incrementally evolved towards a learning management tool. Faculty/instructors may wish to

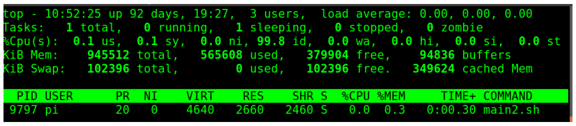

Fig. 5. Front-end script (main2.sh) memory usage is 283.65 KiB in this screen-capture.

Fig. 6. Web browser based, commercial learning center management system memory usage is at 28,656 KiB and is approximately 101 times more memory intensive than the Raspberry-Pi alternative.

track the usage of the learning center and feedback from coaches in certain weeks of complex instruction that have traditionally been challenging for students. This feedback may be used to provide supplementary lectures or study material by the faculty/instructors. Identification of high usage times allows for optimal allocation of resources (tutors). Currently, our engineering learning center works on a fixed limited budget. Resource allocation has always been mostly ad-hoc until now. Data-driven resource allocation will be beneficial.

The cost of the R-Pi based system is significantly lower than the alternative commercial web-based learning center management system. A breakdown of the cost is described in table I for the R-Pi based system. This cost may also be treated as a one-time expense. Future expenses may include purchase of another R-Pi as a back-up or replacement and hardware peripherals (if they were to decay in quality due to use). The memory usage of the BASH script that runs the front-end of the code is signficantly smaller (see figure 5) than that of the web-based commercial alternative (see figure 6).

TABLE I
COST BREAKDOWN OF R-PI BASED SYSTEM

Item	Time invested	Cost per hour or per item	Total cost
Raspberry Pi 3	n/a	$35.00 (per item)	$35.00
PC Monitor	n/a	$179.00[a] (per item)	$179.00
Keyboard	n/a	$15.00[a] (per item)	$15.00
Mouse	n/a	$10.00[a] (per item)	$10.00
HDMI Cables	n/a	$7.00[a] (per item)	$7.00
Lan Cable	n/a	$7.00[a] (per item)	$7.00
Network activation	n/a	$50.00 (one time fee)	$50.00
Programming	32	$40.00[a] (per hour)	$1280.0
Total Cost			**$1583.00**

[a]These items were already available but a cost from Amazon was used to provide an estimate of overall cost that may be incurred if they needed to be purchased.
[b]An inflated cost-per-hour for the "programming" task was utilized.

VI. CONCLUSION, IMPROVEMENT OVER EXISTING SYSTEM AND FUTURE DIRECTION

A Raspberry-Pi based learning center management system, with data processing automation scripts, smaller physical and memory footprint than the commercial alternative and lower cost than the commercial alternative has been successfully deployed in our engineering learning center. It has operated for a complete semester without outages. The data collection, analysis and visualization are carried out automatically every

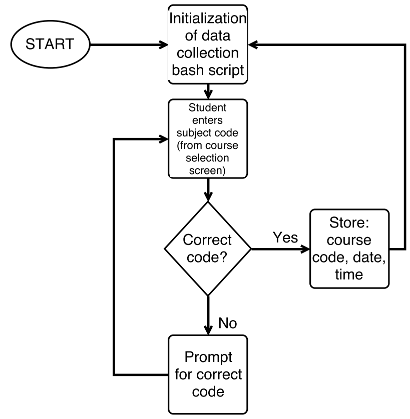

Fig. 7. Course selection screen (fig 1) and it's operation. This screen is run though a bash script that runs perpetually. Once a user selects/enters correct course code, from those available, data is stored as a .csv file and the screen reloads for the next user.

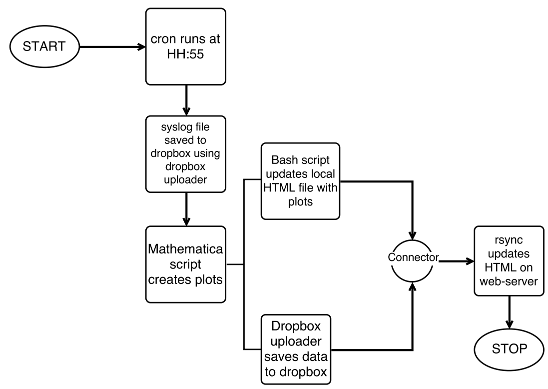

Fig. 8. This background process uses cron, rsyn, dropbox uploader [13] to manipulate/operate on data collected and to create a set of plots (using Mathematica) that are updated on a HTML dashboard for learning center administrators.

hour and results are made available via a simple HTML based web-page through dashboards. This ability and potential for extensibility is absent from the commercial web-based, academic center management system. Preliminary results generated by our automated scripts, show trends in learning center usage and the courses driving this usage. This information allows us to make informed decisions on staffing the learning center to target high-traffic hours and courses that have a high demand. Future efforts will be to: (1) continue to develop this project to increase the capability of this learning center management system to encompass learning management and time-clock abilities. (2) deploy this across the entire university campus to understand cumulative learning centers' engagement.

ACKNOWLEDGMENT

The author thanks the department of mechanical engineering-engineering mechanics at Michigan technological university for providing funds for the inception and deployment of this project in our engineering learning center.

References

[1] WW Predebon and P Cho. Engineering learning center coach training program for minority students. In *ASEE annual Conference & Exposition*, 1997.

[2] Norma Henderson, M Sami Fadali, and Jerry Johnson. An investigation of first-year engineering students' attitude toward peer-tutoring. In *Frontiers in Education, 2002. FIE 2002. 32nd Annual*, volume 2, pages F3B–F3B. IEEE, 2002.

[3] Jennifer M Good, Glennelle Halpin, and Gerald Halpin. A promising prospect for minority retention: Students becoming peer mentors. *Journal of Negro Education*, pages 375–383, 2000.

[4] Alexander W Astin. Student involvement: A developmental theory for higher education. *Journal of college student personnel*, 25(4):297–308, 1984.

[5] Alexander W Astin et al. *What matters in college?: Four critical years revisited*, volume 1. Jossey-Bass San Francisco, 1993.

[6] Ernest T Pascarella and Patrick T Terenzini. *How college affects students*, volume 1991. Jossey-Bass San Francisco, 1991.

[7] Kenneth A Bruffee. Peer tutoring and the "conversation of mankind.". *Landmark essays: Writing centers*, 9:87–98, 1995.

[8] Beasley, C., 1997. Students as teachers: The benefits of peer tutoring. *Learning through teaching*, pp.21-30.

[9] George Siemens and Phil Long. Penetrating the fog: Analytics in learning and education. *EDUCAUSE review*, 46(5):30, 2011.

[10] Oliver Scheuer and Bruce M McLaren. Educational data mining. In *Encyclopedia of the Sciences of Learning*, pages 1075–1079. Springer, 2012.

[11] Ryan Shaun Baker and Paul Salvador Inventado. Educational data mining and learning analytics. In *Learning analytics*, pages 61–75. Springer, 2014.

[12] Cristobal Romero and Sebastian Ventura. Educational data mining: A survey from 1995 to 2005. *Expert systems with applications*, 33(1):135–146, 2007.

[13] Fabrizi, Andrea. "BASH Dropbox Uploader." , http://www. andreafabrizi. it.