Using Students' Affective State as a Measure of CS Lab Risk in an Early Detection System

Mireilla Bikanga Ada
School of Computing Science
University of Glasgow
Glasgow, United Kingdom
Mireilla.BikangaAda@glasgow.ac.uk

Gareth Sears
School of Computing Science
University of Glasgow
Glasgow, United Kingdom
garethsears@gmail.com

***Abstract*—This paper presents a dual dashboard early warning system which uses students' affective state as a measure of risk. Affective state has been shown to influence CS1 performance, and specific states such as frustration have been linked to attrition. The software administers affective surveys to students using a series of 2-dimensional grids. Students then complete a qualitative journal entry. Risk weights are assigned to students based on the journal response's sentiment analysis and whether student's 2-dimensional grid responses fall within configurable '*danger zone*' bounds. The early warning system automatically flags students as needing support if the responses' combined risk weights exceed configurable thresholds. Additionally, flags can be assigned manually, either by instructors or by students themselves.**

Keywords— CS1, student support, sentiment analysis, early warning system, retention, higher education, student emotions

I. INTRODUCTION

Considerable research has been undertaken to identify which factors influence a student's decision to remain in computing science introductory programming course (CS1) [1]. Though the reasons for non-retention are complex and multifaceted, common themes have emerged when interviewing departing students. These include a perceived lack of student-to-faculty interaction, delays in identifying students at risk, and failures to intervene at critical junctures during the course [1]. Rapid identification of students at risk is paramount in CS1. Students often only realise they have fallen behind when they receive a substantial assignment or a failing mark and may only seek help until at a stage where interventions are unable to make a significant difference [1]. There has been an interest in developing software early warning systems (EWS) to support instructors to this end. These systems' data sources, which include virtual learning environment (VLE) data, assessment scores, institutional variables, differences in technology use, students' online engagement and course design, can be used to predict student retention.

On the other hand, student affect and its influence on attrition and performance have recently gained traction in CS education research. Affective data may offer such an avenue, either as an alternative or a supplement to existing EWS data sources. This paper presents a web application, a student-facing and instructor-facing dual dashboard system which will administer affective surveys to CS1 students following course labs. It will use this data to drive an EWS which automatically identifies students needing support in a sufficiently short time frame. The following section provides the background, including a brief overview of similar applications; Section III presents the architecture and framework used to develop the system; Section IV presents the interface; Section V presents a brief conclusion and future work.

II. BACKGROUND

A. *Surveying emotional response to CS1 components*

There is a rising interest in monitoring students' emotions in CS education research. A subset of this research has focused on tracking students' affective response to specific components of CS1 courses, such as labs and assignments. These studies aim to identify learning barriers and assist with interventions. The vast majority have focused on programming labs and assignments [2, 3,4]. Several of these studies have indicated a correlation between a student's affective state and their achievement. McKinney and Denton found that affective states such as interest/enjoyment, perceived pressure, and perceived competence (among others) significantly correlated with students' CS1 programming grades [2]. They also discovered that students' positive affective states tended to decline over the duration of the course, but this could be tempered with timely affective interventions [2]. Lishinki et al. found students' emotional reactions to programming projects had significant effects on future project outcomes. Feeling frustrated or inadequate had a particularly lasting negative impact [5]. Finally, Haden et al. investigated student affect in programming labs and found that students' satisfaction, confidence with planning, and sense of improvement significantly correlated with their final marks. However, unlike McKinney and Denton, they found that attainment did not correlate with interest, perceived difficulty, or familiarity with the material [3]. Haden et al. used 2-dimensional grids which bound affective criteria together in pairs. There is precedent for this novel approach, as affective state co-occurrence has been documented in previous programming studies [6]. However, it remains unvalidated, and the pairs chosen were based only on instructor intuition [3]. The above studies suggest there is merit to using affective feedback data to direct interventions.

B. *Sentiment Analysis in education*

Sentiment analysis is "the process of identifying and classifying users' opinions from a piece of text into different sentiments - for example, positive, negative, or neutral - or emotions such as happy, sad, angry, or disgusted to determine the user's attitude toward a particular subject or entity" and it is achieved using "machine learning and natural language processing techniques" [7]. Educational sentiment analysis tools uptake is slow. In their literature review of educational sentiment analysis tools, Rani and Kumar cite a few studies that use course feedback surveys as a data source, including feedback from a student response system and student diary entries.

C. Learning Analytics and Dashboards

Learning analytics forms the basis of many early warning systems. Learning analytics tools can benefit stakeholders across all levels of education and use a variety of methods for achieving their goals. Primary techniques include the distillation of data for human judgement, prediction, and outlier detection [8]. LA data can be displayed on dashboards. Dashboards which display identical, or near-identical, information to both students and instructors are known as dual-dashboards. These enable transparency and help address power imbalances between stakeholders [8], alleviating student concerns over invasive observations and reduced autonomy [51].

D. Existing systems

The Student Affect Tool is a series of 2-dimensional XY grids which ask students to evaluate their programming experience in labs against pairs of affective criteria. The tool provides instructors with several data visualisations. Scatter diagrams show answer distributions for each lab's grid and line charts illustrate how students' affective responses vary over the course of a semester. The software aims to provide instructors with actionable data to support early interventions with students experiencing difficulties [4]. However, it does not have a dual dashboard which offers students support status notifications and would address power imbalances. Secondly, the inclusion of qualitative questions may support instructors' interpretation of the data, as the tool's authors state that further student interviews are currently needed to confirm the accuracy of their judgements [4].

StudentsAtRisk is a simple early detection system which flags students who have not engaged with course materials for two weeks as "at risk". It features a dual-facing dashboard as well as the ability for instructors to manually flag students and students to flag themselves [9] manually. This implementation has many positives. Firstly, it empowers students. Its dual dashboard offers transparent representations of data to both instructors and students, and its novel "self-flag" feature promotes student agency [9]. However, its simplicity may not represent the complex and multidimensional factors determining student success [8], pedagogy and adapt thresholds to varying course content and cohorts.

Qualtrics XM is a commercial survey platform that can perform sentiment analysis on survey text responses and classify comments by topic. A drawback of Qualtrics's sentiment analysis is that the corpora used to train its machine learning model is unknown. Rani and Kumar state that mismatched domains can result in erroneous sentiment analyses [7].

III. Framework and System Architecture

The Symfony 5 web framework was chosen as middleware to facilitate development. Symfony broadly follows a model-view-controller (MVC) architecture mediated by its HTTP Kernel interface. Symfony supports rapid development by combining generic modules with a standardised structure [10]. The PostgreSQL relational database was chosen over alternatives in anticipation of the learning dashboard's complex reporting. For example, PostgreSQL can aggregate survey data over time frames using window functions which are not available in noSQL or lightweight relational alternatives (e.g. MySQL). MonkeyLearn's Text Analyser API was chosen for sentiment analysis [11], as training and validating a context-sensitive model would require time. The application uses Symfony's Twig 3 template engine for the view layer, which can interpolate dictionaries of PHP objects into HTML responses. The application's security builds on Symfony's Security Bundle modules. Figure 1 presents the system architecture of the web application.

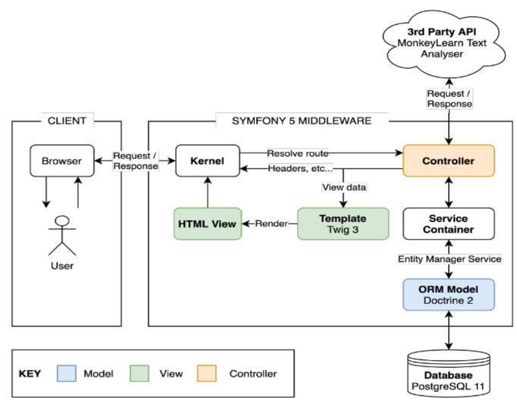

Figure 1. System Architecture

A. Risk Calculation

The proposed solution assigns risk weights to survey item responses which fall within their corresponding question's danger zone value bounds. XY questions' danger zones correspond to quadrants on the response grid, and sentiment questions' danger zones correspond to confidence bounds for negative classifications. The students' combined risk weights for a lab is termed their risk factor. Risk factor calculation is outlined in Figure 2. Students would be automatically flagged as needing support if their lab risk factor exceeded *X*% for *Y* consecutive weeks.

$$riskfactor(\%) = \frac{\sum_{i=1}^{n} r_i}{\sum_{i=1}^{n} rmax_i}$$

where n = survey item index

r = risk weight assigned to student's i$^{\text{th}}$ item response

$rmax$ = the maximum risk weight possible for the i$^{\text{th}}$ survey item

Figure 2: Risk factor calculation

IV. INTERFACE

A. Dual Dashboard

The dashboard enables instructors to see a summary of all students at risk on a course, review an individual student's survey responses, while students can review their own risk. Students can see graph summaries of their own lab responses over the course. Instructors can see all graph summaries. Figure 3 presents the student dashboard, which is identical to that of the instructors except for their respective message.

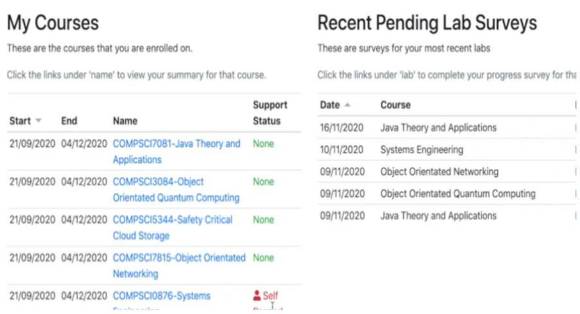

Figure 3. Student dashboard

B. Survey Instrument

Once students click on a pending lab, they are provided with an XY interface for querying their affective responses, as seen in Figure 4. The students can also enter qualitative feedback on their course, which is analysed using sentiment analysis tools.

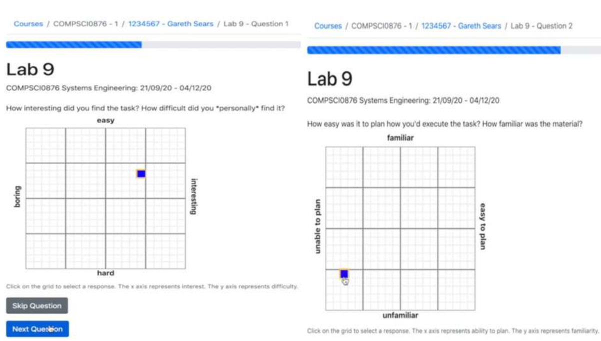

Figure 4: Survey instrument on the system – Student view

C. Early Warning System

The EWS calculates a risk factor for students based on their lab survey responses and automatically flag students at risk at regular time intervals on a course-by-course basis. A student can see that they have been flagged and can flag themselves, and the instructors can flag them. The instructors can define and configure danger zones for survey responses (see Figure 5) and can configure the course thresholds for risk.

V. CONCLUSION

This paper introduced a prototype application that can help identify students at risk of falling behind their labs using their affective state. The system allows the students to evaluate their labs every week and enable instructors to monitor the students. Future work will include a complete evaluation of the system and explore if the students understand and are able to interpret the meaning the dashboard generated.

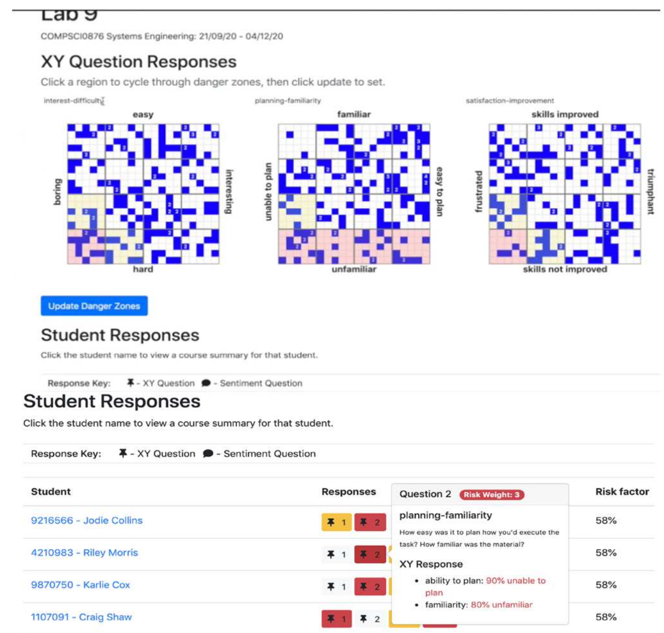

Figure 5. XY survey response and sentiment feature on instructors' dashboard

REFERENCE

[1] A. Petersen, M. Craig, J. Campbell, and A. Tafliovich. "Revisiting whystudents drop CS1," *ACM International Conference Proceeding Series*, pp. 71–80, 2016

[2] M. Liz-Domınguez, M. Caeiro-Rodrıguez, M. Llamas-Nistal, and F. A. Mikic-Fonte. "Systematic literature review of predictive analysis tools in higher education," *Applied Sciences* (Switzerland), vol. 9, no. 24, 2019.

[3] P. Haden, D. Parsons, J. Gasson, and K. Wood. "Student affect in CS1: Insights from an easy data collection tool," *in ACM International Conference Proceeding Series*. Association for Computing Machinery, New York, USA, pp 40–49, 2017.

[4] D. Parsons, K. Wood, J. Gasson, and A.C.M. Moskal. "Development of a Self- Reporting Tool for Capturing Student Emotions during Programming Activities," *in ACM International Conference Proceeding Series*. . Association for Computing Machinery, New York, USA, pp. 64–68, jan 2019.

[5] A. Lishinski, A. Yadav, and R. Enbody. "Students' emotional reactions to programming projects in introduction to programming: Measurement approach and influence on learning outcomes," *in ICER 2017 - Proceedings of the 2017 ACM Conference on International Computing Education Research*, New York, USA, pp. 30–38, aug 2017.

[6] N. Bosch and S. D'Mello. "The Affective Experience of Novice Computer Programmers," International Journal of Artificial Intelligence in Education, vol. 27, no. 1, pp.181–206, 2017.

[7] S. Rani and P. Kumar, "A Sentiment Analysis System to Improve Teaching and Learning," in *Computer*, vol. 50, no. 5, pp. 36-43, May 2017, doi: 10.1109/MC.2017.133.

[8] P. Leitner, M. Khalil, and M. Ebner. "Learning analytics in higher education - a literature review," *in Studies in Systems, Decision and Control*, vol. 94, pp. 1–23, Springer International Publishing, 2017.

[9] M. Bikang Ada and K. Turinicova, "Developing a Dual Dashboard Early Detection System," 2020 IEEE 20th International Conference on Advanced Learning Technologies (ICALT), Tartu, Estonia, 2020, pp. 155-157, doi: 10.1109/ICALT49669.2020.00052.

[10] SensioLabs. What is Symfony? https://sensiolabs.com/

[11] MonkeyLearn. Monkeylearn - text analysis. https://monkeylearn.com/