

Knowledge Acquisition for Learning Analytics: Comparing Teacher-Derived, Algorithm-Derived, and Hybrid Models in the Moodle Engagement Analytics Plugin

Danny Y.T. Liu^{1,2(✉)}, Deborah Richards², Phillip Dawson³,
Jean-Christophe Froissard⁴, and Amara Atif²

- ¹ Educational Innovation Team, The University of Sydney, Sydney, Australia
danny.liu@sydney.edu.au
- ² Faculty of Science and Engineering, Macquarie University, Sydney, Australia
{danny.liu,deborah.richards,amara.atif}@mq.edu.au
- ³ Centre for Research in Assessment and Digital Learning,
Deakin University, Melbourne, Australia
p.dawson@deakin.edu.au
- ⁴ Faculty of Arts, Macquarie University, Sydney, Australia
chris.froissard@mq.edu.au

Abstract. One of the promises of big data in higher education (learning analytics) is being able to accurately identify and assist students who may not be engaging as expected. These expectations, distilled into parameters for learning analytics tools, can be determined by human teacher experts or by algorithms themselves. However, there has been little work done to compare the power of knowledge models acquired from teachers and from algorithms. In the context of an open source learning analytics tool, the Moodle Engagement Analytics Plugin, we examined the ability of teacher-derived models to accurately predict student engagement and performance, compared to models derived from algorithms, as well as hybrid models. Our preliminary findings, reported here, provided evidence for the fallibility and strength of teacher- and algorithm-derived models, respectively, and highlighted the benefits of a hybrid approach to model- and knowledge-generation for learning analytics. A human in the loop solution is therefore suggested as a possible optimal approach.

Keywords: Learning analytics · Goal seeking · Human and machine models · Hybrid models · Knowledge acquisition

1 Introduction

In the era of Big Data where knowledge is machine generated from raw data, one might assume that the knowledge acquisition bottleneck identified in 1980 by Feigenbaum [1] that relies on humans to articulate their knowledge has been overcome. This bottleneck was seen to be the major impediment to the advancement of artificial intelligence and highlighted the difficulties associated with expressing and representing what human experts “know” [1]. Challenges included gaining access to domain experts willing and

able to articulate their knowledge and, if this hurdle was overcome, having to understand and encode the terms and concepts in often unfamiliar domains and find a way to validate and maintain the resultant knowledge base(s) [2].

Data-driven approaches to knowledge discovery may seem to circumvent the knowledge engineering process and the need for knowledge acquisition from humans, but even advanced data processing methods or algorithms often include one or more steps that rely on human input, for example to set weights or identify cases and features. Even where human input is not essential and machine-learned weights and features can be used, inclusion of humans at key points in the process can significantly reduce the search space and improve performance [3]. As with any computer software, verification can be done by a machine, but validation requires human involvement to confirm that it is fit for purpose and addresses needs. Similarly, measures of interestingness and other machine-derived indicators will not be able to replace the domain expert when it comes to validation and utilization of the knowledge uncovered. We believe this is particularly true in the subfield of learning analytics, where the activities undertaken by learners have been designed by the domain expert (teacher) to aid learning and thus measurement of the learner's performance in these activities is fundamentally tied to the domain knowledge and pedagogical strategy (or problem-solving knowledge) of the teacher [4].

Together with other researchers, we have developed a tool, known as the Moodle Engagement Analytics Plugin (MEAP), that uses data mining methods on student data associated with their use of the popular Moodle learning management system (LMS), to identify levels of student engagement in a course and determine a risk rating that could predict success or failure in a course [5–7]. To be able to calculate the risk rating it is necessary to acquire course-related knowledge from the teacher in the form of the salient features (or triggers) that indicate engagement and participation, and the thresholds (or parameters) and weightings of each of these features that may indicate risk. Similar to the knowledge acquisition bottleneck, teachers have found providing these parameters and weightings a barrier to adoption of the system because assigning them is time-consuming and hard to articulate and validate, leading to a lack of confidence in the risk rating produced.

To address this barrier, in this paper we present a preliminary exploration of the use of machine learning on historical data to generate the parameters and weightings. We correlate the results of the human teacher-derived models with machine (algorithm)-derived models against actual student performance data to determine which performs better. We also investigate the performance of 'hybrid' models.

In the next section we introduce the architecture of MEAP. In Sect. 3 we present our methodology including the courses chosen for our evaluation, how teachers assigned parameters and weightings, the algorithm used to determine these, and how to compare the different models. Results appear in Sect. 4, followed by discussion in Sect. 5, and conclusions and future directions in Sect. 6.

2 The Moodle Engagement Analytics Plugin

The Moodle Engagement Analytics Plugin is open source software that plugs into the Moodle LMS¹. It is built around ‘indicators’ that each address an aspect of students’ expected engagement with a course. The three primary indicators are assessment, forums, and logins, in keeping with literature that highlights these metrics as informative for student engagement and performance [8–10]. These indicators read data from the Moodle database and, based on user-defined parameters and weightings, are able to calculate a risk rating for each indicator for each student in a course (Fig. 1). These risk ratings for the individual indicators are then weighted by user-definable weightings, and form a total risk rating for each student which is then reported by the tool (Fig. 1). Previous work has validated the efficacy of the total risk rating to reflect student course performance, and therefore provide a useful measure of student engagement and disengagement [6].

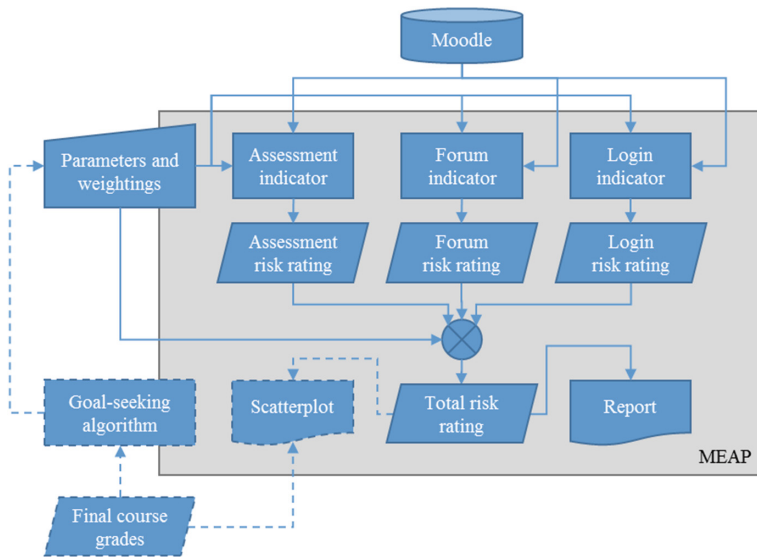


Fig. 1. Architecture of the Moodle Engagement Analytics Plugin (MEAP). Additional structures added for this study indicated by dashed outlines.

The original release of MEAP allowed teachers to self-define the parameters and weightings for the tool, ostensibly using their domain knowledge and experience. To enable the current study and allow teachers to leverage machine knowledge derived from the actual Moodle dataset, we added functionality into MEAP that could algorithmically determine optimal parameters and weightings using these data (Fig. 1).

¹ Available at https://github.com/netspotau/moodle-report_engagement as open source software.

What follows is a quantitative analysis of such teacher-derived and algorithm-derived knowledge in the context of this learning analytics tool.

3 Methodology

3.1 Course Selection

We present a preliminary examination of three undergraduate courses taught at Macquarie University. Each course was delivered in 2014, and again in 2015. There were two humanities courses (first year, HUM1, and second year, HUM2) and one first year science course (SCI1) (course codes have been deidentified). These courses were examined due to their relatively high failure rates (15-26 %) and sizeable classes (Table 1), making them important for analysis [6]. This study was approved by the Macquarie University Human Research Ethics Committee (approval numbers 5201300866 and 5201500031). Table 1 provides a summary of each course's relevant Moodle learning designs, with a focus on aspects most relevant (and accessible) to MEAP.

3.2 Teacher-Derived Models of Knowledge

One teacher (the course coordinator) was interviewed per course early in the semester that the course was offered in 2015. Teachers were first given a basic introduction to

Table 1. Outline of courses examined in this study. Activities falling outside the study timeframe (up until the end of week four; see Sect. 3.4) or not captured online are denoted in square brackets

Course	Size	Relevant course designs in 2014 offerings
HUM1	151	<ul style="list-style-type: none"> Assessments: weekly online quiz, [two essays, pre-readings, class participation] Forums: announcements, general discussion, welcome/introductions Well-structured Moodle site with readings, links, lecture slides and recordings, assessment guidance
SCI1	57	<ul style="list-style-type: none"> Assessments: weekly online exercise, diagnostic quiz, assignment, [two assignments, test, and exam] Forums: announcements and general discussion Well-populated Moodle site with lecture and tutorial notes, assessment submission boxes
HUM2	78	<ul style="list-style-type: none"> Assessments: online discussion, [class participation, essay, exam] Forums: announcements, general discussion, weekly discussion (assessed) Well-structured Moodle site with weekly readings and additional resources, links, lecture notes and recordings, assignment submission boxes

MEAP, which involved the researchers describing the data from the Moodle logs that were analyzed by MEAP, the three indicators (assessment, forum, and login), and how the indicators each produced a risk rating that were combined to form the total risk rating. Teachers were then asked to set the parameters for each indicator and the indicator weightings, by conceptualizing what they expected of a good student. They were also interviewed on their conceptions of identifying student engagement (e.g. what might be effective variables to measure engagement and performance), and any challenges they perceived in using the tool.

3.3 Algorithm-Derived Models of Knowledge

Algorithm design. We developed a simple goal-seeking algorithm and embedded it into MEAP. The algorithm was loosely based on a simulated annealing approach [11], where parameters would be iteratively adjusted to improve the outcome of an acceptance function, which was, in this case, maximizing the inverse correlation between total risk rating and final course grade for each student (i.e. a correlation closer to -1 was preferred, as total risk rating should be inversely correlated with final course grade, used here as a proxy for student performance). Pseudocode for the goal-seeking algorithm is presented below (Algorithm 1), and the full code is open source².

```

Set array of parameters with starting values
For i = 1 through imax
  Randomly shuffle array of parameters
  For each parameter
    For j = jmax through 1
      factor = random number between 1.2 and (j * 1.5)
      new_hi = current_value * factor
      new_lo = current_value / factor
      For each value (new_hi, new_lo, current_value)
        Update indicator settings
        Calculate risk ratings for each student
        Correlate risk ratings with outcome variable
      Select value that yields best correlation
      Set current_value to be that value
Visually report best correlation to user

```

Algorithm 1. Pseudocode for goal-seeking algorithm used to determine optimal parameters.

The algorithm starts with pre-defined starting values and, at each iteration, uses a pseudorandom factor to move each parameter higher and lower, and tests these parameter values to find the best direction to move in to improve the correlation

² https://github.com/dannyliu-mq/moodle-mod_engagement/tree/indicator_helper and https://github.com/dannyliu-mq/moodle-report_engagement/tree/indicator_helper.

between risk rating and the outcome variable (in our case, the final course grade was used as a proxy for student performance). The algorithm was crudely designed so that at each iteration of j , the movement would become more limited as the algorithm approached an optimal solution. Our aim in embedding this goal-seeking algorithm into the existing Moodle plugin was so that teachers could ultimately use it themselves to assist in determining optimal parameters and weightings. This was presented as a graphical user interface (Fig. 2). However, being embedded within a Moodle page imposed some technical limitations on this approach. Most notably, this included a PHP script execution timeout limit (typically 30 s) which, in our technological context, practically constrained the number of steps through which the algorithm could iterate. We are currently working on an alternate approach that uses client-side asynchronous requests to the server for each iteration, mitigating these timeout limitations, in conjunction with a genetic algorithm instead of simulated annealing so that the 28 parameters and weightings can be searched simultaneously.

The screenshot shows a web-based configuration interface with the following elements:

- Target variable**: A dropdown menu with the value 'sng' selected.
- Discovery mode**: A dropdown menu with the value 'By individual selected indicator' selected.
- Indicator to discover parameters**: A dropdown menu with the value 'assessment' selected.
- Indicator(s) to discover weightings**: Three checkboxes, all of which are checked: 'assessment', 'forum', and 'login'.
- Overall iterations**: A dropdown menu with the value '2' selected.
- Iterations for each parameter**: A dropdown menu with the value '3' selected.
- Show data dump**: An unchecked checkbox.
- Run parameter discovery**: A button located at the bottom of the form.

Fig. 2. Screenshot of teacher-facing interface for parameter and weighting discovery

Deriving parameters using the goal-seeking algorithm. To determine the algorithm-derived parameters and weightings for each course, the target outcome was specified as course final grade and the number of steps (i and j ; Algorithm 1) were typically set to 4-5. The algorithm was first run on each indicator (assessment, forum, and login) separately to determine the optimal parameters for each indicator. The algorithm was then run to determine the optimal weightings of each of the three indicators. Finally, the

Pearson correlation coefficient was reported by the tool, expressing the correlation between the total risk rating and final course grade for each student in the course.

3.4 Comparing Teacher- and Algorithm-Derived Models

To determine the correlation between total risk ratings reported by MEAP and final course grade for each student, the parameters and weightings that each teacher determined at the beginning of semester one, 2015, for their course were entered into MEAP after the conclusion of the semester and final course grades were available. The new functionality built into MEAP allowed the reporting of the Pearson correlation coefficient in each course between each students’ course final grade and calculated risk ratings. These correlations were calculated for each separate indicator (assessment, forum, and login), as well as for the total risk rating.

The same method was followed to determine the correlations between course final grades and risk ratings as calculated according to algorithm-derived parameters and weightings. An overview of this process is presented in Fig. 3. Because previous work on MEAP identified week four (out of a 13-week semester) as a best compromise between early detection and the availability of sufficient data [6], we limited MEAP to only use Moodle data available from the beginning of semester up until the end of week four, in each of the offerings of the courses analyzed.

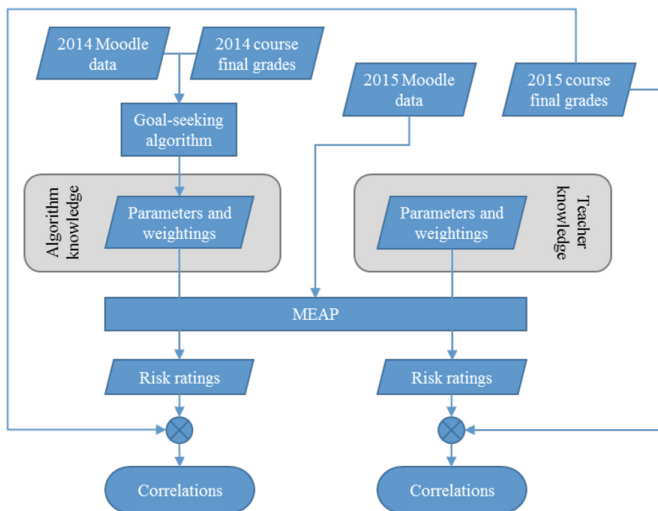


Fig. 3. Overview of knowledge discovery and application process. Algorithm knowledge was derived through goal-seeking analysis of previous (2014) course offering data, while teacher knowledge was derived from human understanding of each course. Both models were tested by generating risk ratings through the Moodle Engagement Analytics Plugin (MEAP) and correlating these with course data from 2015.

4 Results

4.1 Teacher- and Algorithm-Derived Models

The parameters for each individual indicator (assessment, forum, and login), as well as the overall weightings of each indicator, are presented in Table 2 as determined by the teacher of each course, and by the goal-seeking algorithm within each course. In some situations, parameters and weightings were differentially ignored by teachers or the algorithm; that is, their weighting (or importance) was set to 0 % and therefore had no impact upon risk rating calculation.

Teacher-derived models. For HUM1, the teacher's conception of what was important in the course impacted the settings they determined, although they acknowledged that this may differ to how students perceive the course: *"At a glance I can basically already highlight, for instance, that level of engagement and basically whether they're logging into [the LMS]... I've given you certain weightings on what I think is important for them but maybe my expectations of them might not be realistic"*. This was reflected in their preference for the login indicator, and the length of each session (Table 2).

Similarly, the teacher of SC11 had firm beliefs regarding the prevailing indicator, in this case assessment: *"I think participation in - or I should say submission of the assessment tasks... This is quite an effective way of measuring performance."* Again, this was reflected in their chosen weightings (preferring the assessment indicator), and the lack of leeway given to late submissions (Table 2).

For HUM2, the teacher specified that *"the online discussion is crucial and so if students aren't involved on the online discussion early on they fall behind very quickly in terms of understanding concepts."* Perhaps reflecting the nature of the discussions being online (and therefore students needing to log in to access them), this teacher more evenly balanced the forum and login indicators and placed emphasis on new posts and reading posts (Table 2).

Algorithm-derived models. The parameters and weightings determined by the goal-seeking algorithm to optimize the correlation coefficient are also presented in Table 2. There was a stark contrast between the weightings determined by the teacher and those determined by the algorithm, especially in SC11 and HUM2. Indicator-specific parameters were also variable, with little agreement between teacher- and algorithm-derived models. There was also no appreciable pattern of difference between these two models.

4.2 Predictive Power of Teacher-Derived, Algorithm-Derived, and Hybrid Models

The correlations between final course grades and risk ratings reflected this dissonance between teacher- and algorithm-derived models. The scatterplots in Fig. 4 show the correlation between final course grade for each student and their corresponding risk rating for a particular indicator (login) in HUM1. Although the teacher-derived model (generated in 2015 and applied to the 2015 course offering) is correlated in the right

Table 2. Parameters and weightings determined by teachers (T) and the algorithm (A) for the three courses examined in this study. Dashes indicate ignored parameters or weightings

Variable		HUM1		SC11		HUM2	
		T	A	T	A	T	A
Overall indicator weightings	Assessment (%)	30	14	60	–	15	1
	Forum (%)	10	7	10	100	45	89
	Login (%)	60	79	30	–	40	10
Assessment indicator	Grace days	0	2	0	6.4	0	0
	Maximum days	5	3.2	0	7	5	14
	Submitted weighting (%)	50	50	50	0	50	50
	Non-submitted weighting (%)	100	39	100	100	100	100
Forum indicator	New posts per week (no risk, max risk, % weighting)	1, 0, 40	–	0.5, 0, 12	2, 0, 27	1, 0, 40	0.05, 0, 26
	Read posts per week (no risk, max risk, % weighting)	3, 0, 40	0.6, 0, 77	1, 0, 12	0.8, 0, 46	1, 0, 40	4.3, 0, 38
	Replies per week (no risk, max risk, % weighting)	3, 0, 30	0.3, 0, 9	1, 0, 20	–	1, 0, 20	0.5, 0, 23
	Total posts per week (no risk, max risk, % weighting)	–	0.6, 0, 14	1, 0, 56	2, 0, 27	–	0.3, 0, 13
Login indicator	Logins in the past week (number, % weighting)	4, 15	34, 29	–	4, 32	3, 25	4, 11
	Logins per week (number, % weighting)	4, 15	6, 36	2, 50	13, 15	3, 25	2.3, 66
	Length of login (minutes, % weighting)	20, 60	5, 35	30, 50	3.8, 2	20, 25	53, 23
	Time since last login (days, % weighting)	2, 10	–	–	14, 51	7, 25	–
	Session length (minutes)	20	60	60	60	20	60

(negative) direction, it is non-significant and has a low effect size [12]. In comparison, the algorithm-derived model (generated from 2014 data and applied to the 2015 course offering) had a medium-large effect size and was highly significant.

In all three courses examined, the algorithm-derived model outperformed the teacher-derived model when the overall risk rating was correlated with final course grade (Table 3 and Fig. 5). This pattern was also reflected for the login indicator, where the correlation coefficients were consistently closer to -1 in the algorithm-derived models (Table 3). In fact, the teacher-derived models for the login indicator in SC11 and HUM2 led to positive correlation coefficients, meaning that the indicator

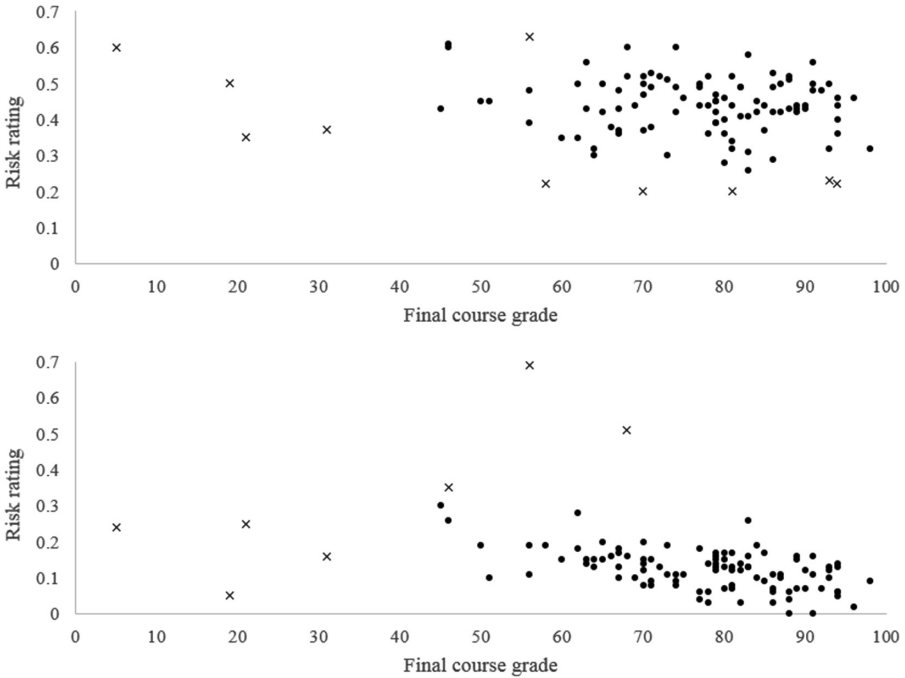


Fig. 4. Correlations between risk rating and course final grade based on human-derived (top) and machine-derived (bottom) parameters for the login indicator in HUM1. Crossed points were considered outliers (Z-score greater than 2). Correlation coefficients are -0.1001 (top, $p > 0.30$) and -0.4271 (bottom, $p < 0.0001$).

would be associating high-performing students with higher risk ratings. The higher power of the algorithm-derived models was also seen for the assessment indicator, although not to the same extent (Table 3). The zero correlation coefficient in SCI1 for the assessment indicator was a product of the calculations that MEAP performs and the teacher-derived setting of ‘0’ for ‘maximum days’ (Table 2). This actually nullified the assessment risk calculation due to the internal algorithmic design of the assessment indicator, which scales the risk calculation from zero days to the ‘maximum days’ setting. This possibly reflects a misunderstanding by the teacher of the somewhat opaque underlying mechanisms of risk rating calculation. In terms of the forum indicator, the teacher- and algorithm-derived models had similar power (Table 3).

To examine the power of a crude hybrid model, we took the mean of teacher- and algorithm-derived parameters and weightings and used these to update the MEAP settings and calculate risk ratings, and then the correlations between these and course final grades. Interestingly, in some instances (HUM1) the hybrid model underperformed the algorithm-derived model, but in SCI1 and HUM2 the hybrid model outperformed both teacher- and algorithm-derived models (Table 3 and Fig. 5).

Table 3. Correlations between course final grades and risk ratings (for individual indicators and overall) as determined from teacher, algorithm, and hybrid models. * $p < 0.5$; ** $p < 0.01$; *** $p < 0.001$. Note: no assessments were detected in HUM2 during the study time period, hence these data were not available (N/A).

		Correlation coefficients			
		Assessment risk rating	Forum risk rating	Login risk rating	Overall risk rating
HUM1	Teacher	-0.493 ***	-0.438 ***	-0.100	-0.172
	Algorithm	-0.494 ***	-0.398 ***	-0.427 ***	-0.379 ***
	Hybrid	-0.494 ***	-0.345 ***	-0.254 **	-0.260 **
SCI1	Teacher	0.000	-0.172	+0.377 ***	+0.300 **
	Algorithm	-0.375 ***	-0.137	-0.031	-0.137
	Hybrid	-0.385 ***	-0.176	-0.032	-0.375 ***
HUM2	Teacher	N/A	-0.245 *	+0.123	-0.224 *
	Algorithm	N/A	-0.276 *	-0.149	-0.247 *
	Hybrid	N/A	-0.239 *	-0.196	-0.326 **

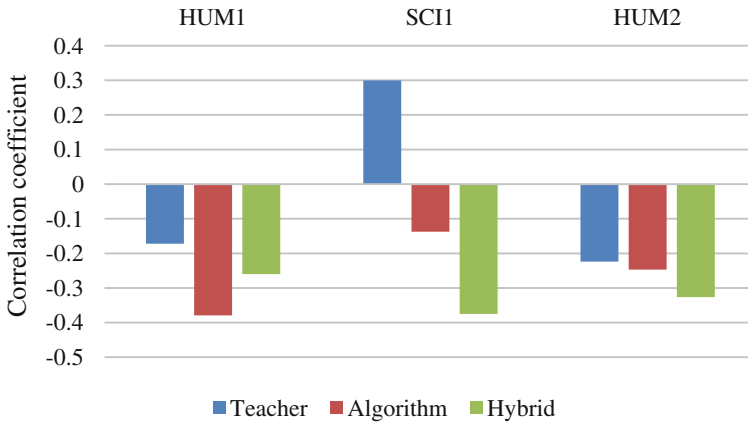


Fig. 5. Pearson correlation coefficients between overall risk ratings and course outcome, comparing the coefficients derived from teacher- and algorithm-derived models and hybrid models. A correlation coefficient closer to -1 indicates a better correlation. (Color figure online)

5 Discussion

The hybrid model can be seen as a rudimentary human in the loop solution. Human in the loop learning can take the form of active learning or learning via queries [13], where the query drives the interaction. Alternatively, the human can drive the interaction as in

the case of intelligent user interfaces or personalization agents, where the human is able to drive the learning while the system observes the human's behavior [14].

Returning to our earlier consideration of whether the knowledge acquisition bottleneck remains a problem, and focusing within the context of learning analytics on student data captured in an LMS, we note that reliance on the human teacher to provide the parameters and weightings can be a major problem. Not only did the teachers in our study report that this was a barrier to usage for them because of the difficulty of coming up with initial parameters and weightings, it was also time-consuming to explore and make sense of the resultant risk ratings and optimize the settings. While in our case the domain expert, rather than a knowledge engineer, was responsible for encoding their own knowledge of their course and their students in the form of assigning indicator parameters and weightings, they faced similar problems to those faced by knowledge engineers. In addition to facing issues similar to limited availability and inability to articulate what they knew, it could also be said that they experienced other reported contributing factors to the knowledge acquisition bottleneck. Ruqian [2] enumerates the following problems faced by the knowledge engineer:

1. They must "extract as much knowledge as possible from the expert's memory and behavior; what the expert provides is only raw material, often mixed with personal biases, even wrong conclusions; this imply the need to screen out, test and reorganize the knowledge obtained from the expert;
2. knowledge is not equal to experience; experience is not always representable; it may be fuzzy and inconsistent; it may appear in the form of inspiration and randomly emerging ideas; the expert has difficulty in explaining it and the knowledge engineer has difficulty in understanding it;
3. there is no clear border between domain knowledge and common sense knowledge; the latter is informal, infinite, continuous and exists everywhere; it is difficult to decide what should be acquired and what should not be acquired;
4. knowledge cannot be acquired at one stroke; it has to be accumulated during a long process; even the most experienced expert is not able to provide this knowledge at a stretch." (p. 2)

Similarly, the teacher needs experience to be able to assign parameters and weightings and make sense of the results. However, just as it has taken time to acquire that knowledge and the knowledge itself is evolving, these settings are mutable and the assignment process is iterative and time-consuming. Moreover, their experience does not necessarily translate into settings that are useful or accurate. This was particularly visible in SCII where the teacher-derived model resulted in a positive correlation between total risk rating and course final grade, implying that the teacher-derived model actually reflected the engagement expected from poor performing students, not high performing ones. In this course for this teacher, an additional complicating factor was a misunderstanding of the impact of certain parameters (notably for the assessment indicator) that further contributed to poor performance of the teacher-derived model. This is related to the idea of model 'comprehensibility' [15], where calculation methods that are relatively opaque to the end user (in this case, the teacher) can obscure the usefulness of learning analytics approaches. Further, it could be said that the teacher's

mental model of the risk rating calculation was flawed, resulting in an unfair (at best) or invalid (at worst) comparison between algorithm and teacher.

Although the three teacher-derived models investigated in this preliminary study consistently underperformed, the algorithm-derived models were not consistent outperformers as may be expected from a goal-seeking approach. At one level, this may be a reflection of suboptimal algorithm choice and design as well as limitations with the technology platform; indeed, genetic algorithms and related evolutionary approaches may be better suited to the optimization problem presented in the MEAP parameters and weightings [16]. However, the inconsistent outperformance of algorithm-derived models may also suggest that data-driven acquisition of knowledge by machines (at least in this instance) may not be completely adequate, and that some human input is necessary. Indeed, the simple hybrid models tested were the best performing in two of the three cases. To our knowledge, this is the first report of a learning analytics approach where human and machine models of student engagement are compared and hybridized, and suggests the importance of expert knowledge supported by data-driven knowledge acquisition processes. This provides some preliminary but cautionary evidence against the preponderance of large-scale learning analytics approaches that rely on purely machine-derived models of student engagement and performance [10, 17, 18], and exemplifies the symbiotic roles of human and machine in knowledge acquisition for learning analytics. A key future research direction for developing human in the loop hybrid models will be to move beyond simply taking the mean of teacher- and algorithm-derived models towards having teachers fine-tuning and adapting algorithm-derived models.

Our findings also provide further supporting evidence for a growing perspective in learning analytics that there is no one model that is suitable for all courses [19]. That is, the knowledge surrounding a course is unique and related to instructional and other contexts. The diversity of teacher-derived models reflects this, as do the range of parameters and weightings determined by the algorithm as best-fitting for each course. We suggest that the analysis we have done, and the correlation and parameter/weighting discovery tool that we have built, could potentially aid teachers to revisit their assumptions and knowledge, and potentially modify their teaching strategies and learning designs accordingly. For example, a teacher could, *de novo*, determine their own parameters and weightings and then see how these impact the correlation between risk rating and final course grade for a previous course offering. Then, they could run the discovery engine and derive parameters and weightings using the goal-seeking algorithm. In this extended knowledge acquisition process, they could then trial hybrid models that combine machine-suggested settings with their domain knowledge and experience and settle on a set of parameters and weightings that optimize the correlation between risk rating and course final grade. They would then apply this to the current course offering, and subsequently apply this iterative cycle to accumulate knowledge over time.

6 Conclusions and Future Directions

We have provided preliminary evidence that suggests expert teacher knowledge for learning analytics can sometimes be outperformed by knowledge derived by data-mining algorithms, and that a hybrid approach may be optimal in some instances. Specifically, we have built such knowledge discovery mechanisms into MEAP, an open source learning analytics plugin for the Moodle LMS. Our results also support the growing trend in learning analytics research that emphasizes knowledge of instructional and other contexts in building accurate models.

As future work, a fully-integrated human in the loop approach that provides an intelligent and adaptive user interface able to guide the teacher in setting parameters and weightings (perhaps via seeding of initial settings if historical data from previous course offerings exist and by visualizing ‘what-if’ scenarios using alternative settings) would accelerate the task of determining optimal models. If historical data are not available, the settings could be seeded based on courses with similar learning designs or other characteristics. This is similar to a case-based reasoning approach where experience from other contexts is applied to knowledge-based systems [20]. Alternatively, an improved algorithmic approach (such as genetic algorithms) may provide more suitable seed settings for such exploration.

Acknowledgements. This work was supported by a Macquarie University Teaching Development Grant. We wish to thank all the staff who worked with us on this study, and our colleagues for many insightful conversations. Initial development and conceptualization of MEAP was supported by a NetSpot Innovation Fund grant, and conducted by a team including Adam Olley, Ashley Holman, Angela Carbone, and others.

References

1. Feigenbaum, E.A.: Knowledge engineering: the applied side of artificial intelligence. *Ann. New York Acad. Sci.* **426**, 91–107 (1984)
2. Ruqian, L.: *New Approaches to Knowledge Acquisition*. World Scientific Series in Computer Science. World Scientific Pub. Co. Inc. (1994)
3. Xu, H., Hoffmann, A.: RDRCE: combining machine learning and knowledge acquisition. In: Kang, B.-H., Richards, D. (eds.) PKAW 2010. LNCS, vol. 6232, pp. 165–179. Springer, Heidelberg (2010)
4. Lockyer, L., Heathcote, E., Dawson, S.: Informing pedagogical action: aligning learning analytics with learning design. *Am. Behav. Sci.* **57**, 1439–1459 (2013)
5. Liu, D.Y.T., Froissard, J.-C., Richards, D., Atif, A.: An enhanced learning analytics plugin for Moodle: student engagement and personalised intervention. In: Reiners, T., von Kinsky, B.R., Gibson, D., Chang, V., Irving, L., Clarke, K. (eds.) 32nd Conference of the Australasian Society for Computers in Learning in Tertiary Education, Perth (2015)
6. Liu, D.Y.T., Froissard, J.-C., Richards, D., Atif, A.: Validating the effectiveness of the moodle engagement analytics plugin to predict student academic performance. In: 2015 Americas Conference on Information Systems, Puerto Rico (2015)
7. Dawson, P.: Analytics block to identify students at risk of disengaging. Paper presented at 2012 Moodle Moot, Gold Coast, Queensland, Australia (2012)

8. Macfadyen, L.P., Dawson, S.: Mining LMS data to develop an “early warning system” for educators: a proof of concept. *Comput. Educ.* **54**, 588–599 (2010)
9. Romero, C., Ventura, S., García, E.: Data mining in course management systems: Moodle case study and tutorial. *Comput. Educ.* **51**, 368–384 (2008)
10. Jayaprakash, S.M., Moody, E.W., Lauría, E.J.M., Regan, J.R., Baron, J.D.: Early alert of academically at-risk students: An open source analytics initiative. *J. Learn. Analytics* **1**, 6–47 (2014)
11. Corana, A., Marchesi, M., Martini, C., Ridella, S.: Minimizing multimodal functions of continuous variables with the “simulated annealing” algorithm. *ACM Trans. Math. Softw. (TOMS)* **13**, 262–280 (1987)
12. Cohen, J.: A power primer. *Psychol. Bull.* **112**, 155 (1992)
13. Schohn, G., Cohn, D.: Less is more: active learning with support vector machines. In: *ICML, San Francisco*, pp. 839–846 (2000)
14. Horvitz, E., Breese, J., Heckerman, D., Hovel, D., Rommelse, K.: The Lumiere project: bayesian user modeling for inferring the goals and needs of software users. In: *Fourteenth Conference on Uncertainty in Artificial Intelligence*, pp. 256–265. Morgan Kaufmann Publishers Inc. (1998)
15. Romero, C., Ventura, S., Espejo, P.G., Hervás, C.: Data mining algorithms to classify students. In: *1st International Conference on Educational Data Mining, Montreal*, pp. 8–17 (2008)
16. Bäck, T., Schwefel, H.-P.: An overview of evolutionary algorithms for parameter optimization. *Evol. Comput.* **1**, 1–23 (1993)
17. Arnold, K.E., Pistilli, M.D.: Course signals at Purdue: using learning analytics to increase student success. In: *2nd International Conference on Learning Analytics and Knowledge*, pp. 267–270. ACM, Vancouver (2012)
18. Krumm, A.E., Waddington, R.J., Teasley, S.D., Lonn, S.: A learning management system-based early warning system for academic advising in undergraduate engineering. In: Larusson, J.A., White, B. (eds.) *Learning Analytics*, pp. 103–119. Springer, New York (2014)
19. Gašević, D., Dawson, S., Rogers, T., Gasevic, D.: Learning analytics should not promote one size fits all: the effects of instructional conditions in predicting academic success. *Internet High. Educ.* **28**, 68–84 (2016)
20. Watson, I., Marir, F.: Case-based reasoning: a review. *Knowl. Eng. Rev.* **9**, 327–354 (1994)