Expert Feature-Engineering vs. Deep Neural Networks: Which is Better for Sensor-Free Affect Detection?

Yang Jiang[1], Nigel Bosch[2], Ryan S. Baker[3], Luc Paquette[2], Jaclyn Ocumpaugh[3], Juliana Ma. Alexandra L. Andres[3], Allison L. Moore[4], Gautam Biswas[4]

[1] Teachers College, Columbia University, New York, NY, United States
yj2211@tc.columbia.edu
[2] University of Illinois at Urbana-Champaign, Champaign, IL, United States
{pnb, lpaq}@illinois.edu
[3] University of Pennsylvania, Philadelphia, PA, United States
{rybaker, ojaclyn}@upenn.edu
aandres@gse.upenn.edu
[4] Vanderbilt University, Nashville, TN, United States
{allison.l.moore, gautam.biswas}@vanderbilt.edu

Abstract. The past few years have seen a surge of interest in deep neural networks. The wide application of deep learning in other domains such as image classification has driven considerable recent interest and efforts in applying these methods in educational domains. However, there is still limited research comparing the predictive power of the deep learning approach with the traditional feature engineering approach for common student modeling problems such as sensor-free affect detection. This paper aims to address this gap by presenting a thorough comparison of several deep neural network approaches with a traditional feature engineering approach in the context of affect and behavior modeling. We built detectors of student affective states and behaviors as middle school students learned science in an open-ended learning environment called Betty's Brain, using both approaches. Overall, we observed a tradeoff where the feature engineering models were better when considering a single optimized threshold (for intervention), whereas the deep learning models were better when taking model confidence fully into account (for discovery with models analyses).

Keywords: Student modeling, feature engineering, deep learning, deep neural networks, affect and behavior detection, Betty's Brain.

1 Introduction

Student modeling assumes a crucial role in the field of Artificial Intelligence in Education (AIED). In recent years, there has been a proliferation of models that can infer complex constructs such as scientific reasoning strategies [1, 2], affect [3, 4, 5], and disengaged behavior [5, 6, 7, 8]. One educational data mining method, commonly used to develop automated models of these types of constructs, is to generate a meaningful set of features from data (i.e. feature engineering). This feature set is then used within

machine learning algorithms to learn the mapping from those features to examples of the construct being modeled, also identified by trained experts [e.g., 2, 3, 4, 5, 7].

Automated detectors using feature engineering have achieved reasonably high success in predicting whether a student is engaged, frustrated, confused, or bored, and whether the student will display related affective states and behaviors [3, 5, 9]. In this approach, ground truth (examples of the construct) is typically collected through classroom observations [5, 10], emote-aloud protocols [4], or self-reports [6]. Theoretically-justified features are then created and utilized to build machine-learning predictive models of affective states and behaviors. The resulting detectors make inferences solely using data from student-software interaction, enabling researchers and educators to explore and detect these constructs scalably and in real time. These affect and behavior detectors have been applied to over a dozen learning environments, and have been found to predict long-term learning outcomes [5, 11, 12, 13]. They can also be integrated in learning environments to provide timely information on when the system should intervene to respond to the students' affect and behavior in real time and reduce negative affective states [4].

However, with the rapid development of deep learning [14], there is an emerging interest and effort in applying deep learning for various problems within student modeling [15, 16, 17, 18]. Deep neural networks have enabled leaps forward in prediction accuracy for models in other domains (e.g., image classification [19]), which has driven recent interest in applying these methods to educational problems. In general, early results have been mixed, with optimism about the potential of deep learning for knowledge modeling and performance prediction [18] giving way to evidence of overstated effectiveness [16], and initial evidence that affect detection could be substantially improved through deep learning [15] transitioning to evidence of the models not working for all populations [20]. As such, the advantages (and disadvantages) of deep neural networks for student modeling are not yet well understood. Therefore, a thorough comparison of deep learning and traditional feature engineering methods is needed in student modeling to determine the strengths and drawbacks of each method.

This paper compares several deep neural network approaches with a traditional feature engineering approach. Specifically, we studied these issues in the context of developing detectors of student affective states and behaviors in an open-ended learning environment for middle school science called Betty's Brain [21]. To our knowledge, this study is the first direct comparison of the two approaches on the same data with a thorough exploration of model types and hyperparameters. The comparison in this paper will lead to a better understanding of the advantages and disadvantages of each approach, including insights into situations where one approach is preferable to the other.

2 Betty's Brain

The Betty's Brain software [21], shown in Figure 1, is an open-ended computer-based learning environment where students learn science and complete challenging scientific tasks by constructing a causal map describing a scientific phenomenon (e.g., climate change, ecosystems, thermoregulation). It adopts the learning-by-teaching paradigm to

help students acquire scientific knowledge and gain cognitive and metacognitive skills. The goal for students in Betty's Brain is to teach a virtual agent, named Betty, about the phenomenon by means of a causal map the students build, where causal relationships (e.g., cold temperature leads to heat loss, as shown in Figure 1) can be represented by a set of concept entities connected by directed causal links.

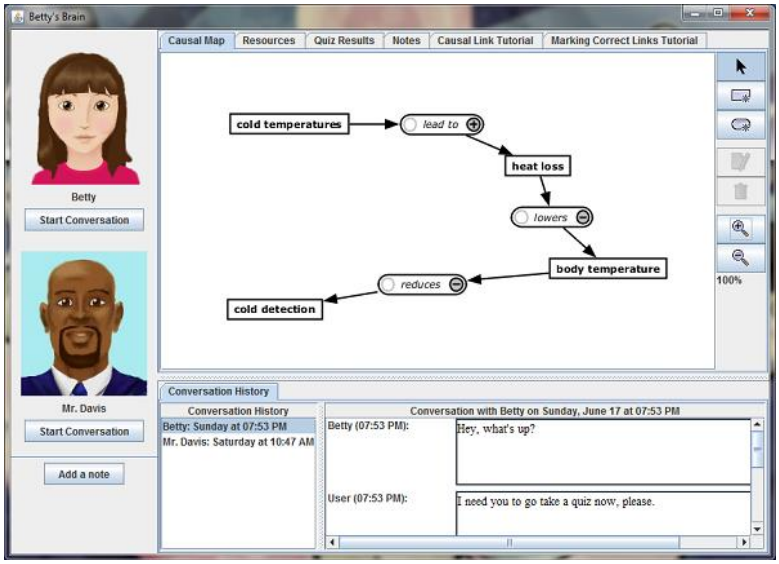

Fig. 1. Screenshot of Betty's Brain.

In this open-ended environment, learners have access to hypermedia resource pages (called the *science book* in Betty's Brain) on relevant scientific concepts to acquire domain-specific knowledge. They can apply what they read about from the resource pages to assist them with the map building. A causal map can be constructed by adding concept entities and creating causal links between specific entities.

Learners can assess their causal map by having Betty, the virtual student, answer questions and explain her answers. Betty's answers to questions are based on the causal map that the student has created, by checking the chain of causal links between the concepts involved in the questions. Students can also request conversations with a pedagogical mentor agent, named Mr. Davis, to evaluate Betty's answer. Additionally, students can have Betty take quizzes (composed of a list of questions to help students improve their causal map) and check the correctness of concepts and causal links and the current state of their causal map, which is compared to the expert model hidden from the system.

Betty's Brain is challenging for students, as it poses high requirements on self-regulated learning. Students need to plan their map construction process, make decisions on when and how to access information pages and which information is important for concept mapping, regularly monitor their causal map by checking Betty's performance,

and accordingly modify their causal maps. These processes, together with the complexity of the task and the open-endedness of the environment, all have the potential to influence engagement and elicit affective and behavioral responses. In this paper, we aim to develop automated detectors of student engagement in the system and compare the accuracy of two sets of detectors respectively using feature engineering and deep learning.

3 Method

3.1 Participants

Participants in this study were a total of 93 sixth grade students from four science classes in an urban public middle school in the southeastern region of the United States. They were observed as they used the Betty's Brain system in spring 2017 and their interactions within the system were logged. The interaction log data and the classroom observations of the students' engagement were used to construct affect detectors.

3.2 Procedure

This study was conducted over a seven-day period. Students took a 30-45 minute paper-based pretest on Day 1 of the study, and received a 30-minute training session on how to use Betty's Brain on the following day. They then spent four class periods working in Betty's Brain to build a causal map about climate change from Days 3–6. They completed a paper-based post-test, which was the same as the pre-test, on Day 7. The pre- and post-tests, composed of multiple-choice items and short response items, were designed to assess students' knowledge of the concepts and the causal relationships underlying the scientific phenomenon in the domain.

3.3 Classroom Observations of Affect and Behavior

While working with Betty's Brain in a classroom setting, students were observed in real-time by two human coders using the Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP 2.0) [10]. BROMP is a momentary time sampling method where students are observed individually, without interruption, in a pre-determined order. BROMP has been applied to explore student engagement by over 150 coders in four countries, resulting in over 25 publications (see review in [10]). It achieves reliably high inter-rater reliability (each of the 150 coders achieved inter-rater reliability with at least one other coder, achieving Cohen's Kappa over 0.6), obtains data quickly, and BROMP data has been used as the basis for a range of automated detectors of affect and engagement [3, 5, 22].

In this study, two BROMP-certified coders observed and recorded affective states (boredom, confusion, delight, engaged concentration, frustration) and behaviors (on-task, on-task conversation, off-task) using an Android application called the Human Affect Recording Tool (HART) [23]. They observed each student consecutively, for up

to 20 seconds at a time, in a predetermined order and recorded the first affective state and behavior that the student clearly displayed during the interval. Observers cycled through the entire class and moved to the next student once the student's state had been determined or 20 seconds expired. These observations were time-stamped and synchronized to the data from the students' interactions with Betty's Brain, which included a total of 146,141 actions generated by students. Early in the study, the coders tested whether they agreed on the affect and behavioral codes and achieved inter-rater reliability (Cohen's Kappa ≥ .60) for affect and behavior.

A total of 5,212 observations of affect and behavior were obtained from the 93 students over the course of this study, with each student being observed 56 times on average across the four class sessions. Engaged concentration was the affective state that was the most commonly identified (4,064 observations; 78.0%), followed by confusion (312 observations; 6%), frustration (241 observations; 4.6%), boredom (220 observations; 4.2%), and delight (149 observations; 2.9%). Student state was marked as *other* if the observer was unsure about the student's affective state or behavior within the 20-second window or the affective state or behavior displayed by the student was not listed in the coding scheme (226 observations; 4.3%).

Behavioral constructs were recorded separately from affect (e.g., a student could be recorded as both bored and off-task). Off-task behavior observations comprised 10.2% (533 observations) of the data, on-task conversation comprised 15.0% (784 observations), and on task behavior comprised 69.0% (3,595 observations).

3.4 Feature Engineering Approach

In the traditional feature engineering approach, we created a set of meaningful features from student interaction log data that could potentially predict specific affective states and behavior. These features were then fed into standard classifiers to train machine-learned predictive models of student affective states and behaviors that were collected via BROMP.

Three broad categories of features were developed for detector construction: 1) basic features, 2) sequence features, and 3) threshold features. The basic features consisted of: a) time-based features which captured the amount of time spent on specific user activities (e.g., total amount of time spent on viewing the causal map; average duration taken each time the student reads a resource), b) frequency/count-based features that calculated the number of times the student executed a specific type of action (e.g., total number of causal links created; total number of notes edited; frequency of moving map elements; number of quizzes student has had Betty take so far), c) ratio/percentage features (e.g., proportion of effective or ineffective actions; concept/link ratio), and d) other descriptive features (e.g., average; standard deviation; minimum and maximum values of map score, which is determined by the difference between the number of correct and incorrect causal links on the map at any point of time). For each feature, we created three variants: a *within 20-second clip*[1] variant, a *thus far* variant, and a *thus far*

[1] 60-second clip variants were initially tested but were less effective than 20-second clips.

divided by time elapsed variant. For example, for the feature "total time spent on viewing causal map," we calculated: 1) the total amount of time student has spent on viewing the causal map within the 20-second observation clip (20-second clip feature), 2) the total time the student has spent on viewing the causal map so far – from the beginning of using the system up to the current time (thus far feature), and 3) the total time the student has spent on viewing the causal map thus far divided by the time that has elapsed thus far (thus far divided by time elapsed feature). In total, 123 basic features were designed and extracted in this study (41 features × 3 variants).

The second category of features involved the frequency of sequences of three consecutive actions. Example three-action sequences included *read resource → add concept → add causal link, read resource → read resource → read resource*, etc. Sequence features captured the frequency of common three-action sequences. First, we searched for all possible three-action combinations executed by students, producing a total of 2,228 possible three-action sequences. Next, we selected the frequent three-action sequences that occurred more than 200 times across all students in the logs from Betty's Brain in order to remove infrequent sequences and obtain a reasonable number of three-action sequences. This reduced the number of sequences to 30. Similar to the basic features, we then applied these sequence patterns to log data and calculated the number of times each sequence occurred within the 20-second clip, the number of occurrences thus far, and the occurrences thus far divided by elapsed minutes. This resulted in a total of 90 sequence features.

We also extracted a set of threshold features that involved selecting an optimized threshold. For example, we determined how *long pause* should be defined in the feature "total number of long pauses after creating causal links thus far." For these features, different thresholds were tested in terms of fit; for example, different thresholds were tried at the grain size of 1 second in order to identify the best threshold for the feature "long pause after building causal links." Thresholds were evaluated based on the correlation between the feature with that threshold and the student's post-test performance. A total of 36 threshold features were generated. Thus, in total, there were 249 features (123 basic features + 90 sequence features + 36 threshold features).

In order to refine the detectors and identify the features most predictive of affective states and behaviors, we adopted a stepwise procedure and tested three sets of features in the final models. First, we constructed detectors using the basic features only. Secondly, we then expanded the feature set and added sequence features to explore the change in model performance. Lastly, we fed all features (basic features, sequence features, and threshold features) into machine-learning algorithms to build detectors. In the following sections, we will discuss the process of building machine-learned models.

Feature Selection. Considering the large number of features we distilled (which increases the risk of over-fitting), especially for the second (basic + sequence features) and third feature sets (basic + sequence + threshold features), tolerance analysis was conducted to reduce the number of features inputted to build affect detectors. Tolerance analysis evaluates the multicollinearity of features and eliminates features that are highly collinear (variance inflation factor > 5).

Forward selection was implemented for further feature selection for each affect and behavior detector, where the feature that most improved model accuracy was added

repeatedly until no more features could be added to improve model performance. Feature selection in this study was conducted within each cross-validation fold and was applied on training data only.

Classification Models. Classification models of affective states and behaviors were constructed in RapidMiner 5.3 [24] in order to determine which features best predict students' affective states and behaviors.

Models were built for each affective and behavioral state using the two-class approach, in which each observation was coded as either the state is present (e.g., bored), or absent (e.g., not bored). For behaviors, we built an off-task detector in order to detect whether the student was off-task or on-task (including on-task conversation). Resampling was implemented (in cross-validated training folds only) using the cloning method in order to make the frequency of each class for each construct balanced.

A small set of classification algorithms that have shown previous success in building affect detectors, including C4.5, RIPPER, Step Regression, Logistic Regression, and Naïve Bayes, were considered and tested for final model.

Detector accuracy was evaluated using two performance metrics: Cohen's kappa and A′. Kappa represents the degree to which the model is better than chance. A detector with a kappa value of 0 performs at chance-level and one with a kappa of 1 performs perfectly. A′ is equivalent to the area under the receiver operating characteristic curve and the Wilcoxon U statistic, and represents the probability that the model can distinguish a positive example (e.g., bored) from a negative example (e.g., not bored). An A′ value of 0.5 indicates chance-level performance and A′ = 1.0 implies perfect performance.

Model performance was evaluated using 10-fold student-level cross-validation. In this process, students were randomly distributed into ten groups. Detectors were trained on nine of the groups and tested on the tenth group. The feature selection was executed on training data only.

3.5 Deep Learning Method

Preprocessing. We converted the event log data to a discrete time series format by coding occurrences of logged actions as 0 or 1 in consecutive three-second intervals. One variable was created for each type of action possible in Betty's Brain. For example, a column was created to denote whether a student was viewing the biology textbook material. If the student quickly browsed the textbook starting 5 seconds into the learning session and viewed for 4 seconds, the time series data for that variable would consist of 0, 1, 1, 0, etc. to capture textbook viewing behavior in the 3-6 second and 6-9 second intervals. We removed any variables with standard deviation $\leq .05$, since these variables represented events that rarely occurred and were less likely to be useful indicators of affect or off-task behavior. Nine variables remained: view causal map, view science book, view notes, view graded questions, view graded question explanation, respond to prompt, add causal map link, move causal map link, and other (context-specific action). Data were then split into sequences of 60 seconds (20 three-second intervals) leading

up to each BROMP observation[2], and two additional variables were added to capture the time since the start of the learning session and the position within the sequences (0 to 60 seconds). This preprocessing method allowed us to create sequences of equal length for each variable, which allows straightforward application of sequential neural network models such as recurrent neural networks.

Deep Learning Model Types. We considered five different common types of neural network models: fully-connected, recurrent neural network (RNN), long-short term memory (LSTM), gated recurrent unit (GRU), and 1-dimensional temporal convolution (Conv). Fully-connected networks consist of layers of simple neurons that connect to every neuron in the previous layer, with no regard to ordering in the sequence. RNNs connect each step in the sequential data to the previous step, thus reducing the number of parameters in the network and allowing the network to learn patterns over time. LSTM networks are a variety of RNN with more complex neurons that include memory cells that can remember elements of the sequence over a long period of time, to capture longer-term dependencies. GRUs are slightly simplified LSTMs that sacrifice some sequence-learning capabilities but require fewer parameters to be learned and thus may work well for smaller datasets. Finally, Conv networks learn filters that match sequences of a specific length (we used length 5). All of our models had an initial hidden layer with 10 neurons (of one of the five different types considered), followed by a fully-connected hidden layer of size 16, followed by an output fully-connected hidden layer of size 2. Exponential linear unit activation was applied after each fully-connected layer [25].

Hyperparameter Selection. Given the small labeled dataset available, a large, complicated network is unlikely to fit well to the data. However, to explore this possibility we adjusted the size of the first hidden layer in increments of 5 neurons from 5 to 70 neurons and added dropout after each layer [26]. We found no notable improvements, and thus continued experiments with no dropout and 10 neurons in the first hidden layer.

4 Results

4.1 Feature Engineering

Performance of the best-performing detector using the feature engineering approach for each construct are shown in Table 1. Prediction models built using the basic features showed better cross-validated performance (kappa and A′ values) for the boredom and delight detectors. This indicated that the count/frequency-based and time-based features were overall better predictors of these affective states than the frequency of action sequences or features that involve threshold fitting. On the other hand, detectors using a combination of feature sets performed better in predicting confusion, engaged concentration, frustration, and off-task behavior.

[2] A 20-second clip was also tested for the deep learning models, but it did not work as well as the 60-second clips.

Overall, all the resulting machine-learned models for these constructs performed better than chance (mean kappa = 0.168, A′ = 0.634), with the detectors for boredom (kappa = 0.278, A′ = 0.682) and off-task behavior (kappa = 0.369, A′ = 0.725) yielding better cross-validated performance than detectors for the other constructs. The performance of these detectors was mildly lower than previously published models of affect in other learning environments [3, 5, 27], and moderately lower than past models of off-task behavior [5].

Table 1. Cross-validated performance of affect and behavior detector using feature engineering and deep learning.

Affect/Behavior	Feature Engineering				Deep Learning		
	Feature Set	Classifier	Kappa	A′	Model	Kappa	A′
Boredom	Basic	Logistic regression	0.278	0.682	GRU	0.103	0.672
Confusion	All	Logistic regression	0.091	0.568	GRU	0.091	0.566
Delight	Basic	Step regression	0.070	0.570	GRU	0.035	0.649
Engaged Concentration	All	Logistic regression	0.142	0.624	GRU	0.138	0.619
Frustration	All	Logistic regression	0.056	0.634	GRU	0.041	0.572
Off-Task Behavior	Basic + Sequence	Logistic regression	0.369	0.725	LSTM	0.268	0.761
Average			0.168	0.634		0.112	0.640

Examination of the features selected in each affect and behavior detector indicated that the features that were predictive of each state were similar in nature and could be grouped into the following categories:

- Frequency of causal map construction or causal map annotation actions (e.g., frequency of deleting entity; frequency of deleting causal link; frequency of marking a causal link correct action)
- Status of causal map (e.g., ratio of the number of concepts remaining in map and the number of causal links remaining in map)
- Note-taking behaviors (e.g., frequency of editing note, frequency of viewing notes)
- Evaluation behaviors (e.g., duration of viewing explanation)
- Resource access behaviors (e.g., duration of viewing science book)
- Conversation request (e.g., number of times requesting a conversation with mentor)
- Threshold features (e.g., number of long pauses after taking quiz; number of long pauses after viewing graded explanation from Betty)
- Sequence features (e.g., frequency of sequence *read resource* → *read resource* → *read resource*; frequency of sequence *read resource* → *read resource* → *add concept*)

- Other (e.g., percent of ineffective actions)

These results indicated that the frequency and duration of relevant actions, especially those that were key to the environment, such as map building, resource accessing, note-taking, requesting conversation, and monitoring and evaluating maps, and the sequence of these actions, are meaningful in predicting affective states and behaviors in Betty's Brain.

4.2 Deep Learning

Several notable patterns of results of the deep learning approach can be seen in Table 1. GRU networks were, on average, the most accurate type across the different detection tasks (mean kappa = 0.110, mean A′ = 0.637). LSTMs were the next most accurate (mean kappa = 0.098, mean A′ = 0.623), demonstrating the efficacy of the simplified structure of GRUs in the current dataset with relatively few instances for deep learning applications — though this pattern was reversed in [15]. The fully-connected network structure was the least accurate (mean kappa = 0.068, mean A′ = 0.577), which is unsurprising given that the fully-connected network does not leverage the sequential nature of the data.

Additionally, there were large differences in the accuracy of neural networks for different detection tasks. Confusion and frustration were particularly difficult to detect (A′ = 0.566 and 0.572 respectively), while boredom was much more effectively detected (A′ = 0.672). Off-task behavior was most accurately detected (A′ = 0.761), indicating that there are clear connections between students' behaviors in Betty's Brain and the BROMP observations of off-task behavior.

Finally, the deep learning models achieved similar or better A′ compared to the feature engineered models for every construct except for frustration, averaging 0.006 higher A′, but achieved worse kappa for every construct except for confusion, averaging 0.056 lower kappa. Examination of ROC curves revealed that the feature engineered models were particularly precise at the pre-selected decision threshold but were less so at other false positive rates, yielding relatively higher kappa values, whereas the deep models were more uniformly effective across decision thresholds.

5 Discussion

The past few years have seen a surge of interest and attention in deep learning [14]. Despite its wide application in other domains such as image classification and natural language processing [14, 19, 28], deep learning is still an emerging area in the field of education with limited studies comparing its predictive power to that of the traditional feature engineering approach. To address this issue, we built predictive models of students' affective states and off-task behavior as they learned science in an open-ended learning environment called Betty's Brain, using both the traditional feature engineering approach and the deep learning approach. Our findings show the advantages and disadvantages of each approach.

5.1 Main Findings

In general, the two approaches yielded similar levels of accuracy, with the detectors using the deep learning approach showing similar or higher A′ (for all but the frustration detector), while the detectors using feature engineering approach generally showing higher kappa (for all detectors except for confusion). These results indicated that the detectors using the feature engineering approach were more accurate in differentiating the presence or absence of an affective state or behavior than the deep learning detectors at one specific threshold (therefore higher kappa); whereas the detectors using a deep learning approach were more accurate in distinguishing whether a student was displaying delight/boredom or not across a wide range of other decision thresholds (therefore higher A′). Such a tradeoff should be taken into consideration when selecting final models. For instance, the deep learning model would be preferable if we want to integrate a detector with a tunable threshold or multiple thresholds, but would be less useful for a single threshold making a single distinction at 50% confidence. The difference in the two metrics for the deep models is also consistent with what was found in Botelho et al. [15], where deep learning affect detectors showed a notable increase in A′ compared to previously published feature-engineered detectors on the same dataset, whereas kappa was slightly lower for the deep learning models.

Another key result, as illustrated by the confusion models, is that deep neural networks are not a panacea. Features engineered to capture confusion were not effective at much above chance levels, and deep neural networks were not able to capture key details researchers may have missed when engineering features. It is possible that students simply do not interact with Betty's Brain in ways that distinguish confusion from other affective states.

Overall, these affect detectors showed relatively lower performance than those constructed in other learning environments such as Cognitive Tutor [3]. This may be due to the open-ended nature of the environment. In many computerized learning systems, student actions are more restricted and each action can be either correct or incorrect based on the answer. In Betty's Brain, however, there is no single correct path. In order to succeed in the environment, students can execute many possible paths, and they have the freedom to decide their own actions at any time. This might make it difficult to create features that capture attributes of the student's learning, which could help identify affective states and behaviors.

In comparing the two modeling approaches to each other, a key advantage of deep neural networks is their capability to automatically derive meaningful features from raw interaction data. Indeed, this is the core capability that has driven advances in deep learning models, and what distinguishes them from shallow neural networks. However, it is also arguable that time saved by this advantage is lost due to time spent refining the structure of neural networks, which is also an open-ended, time-consuming task for any new domain. As such, further research is needed to quantify this tradeoff.

On the other hand, the traditional feature engineering approach has its own strengths. The resulting models using feature engineering, particularly simple models such as logistic regression, are more interpretable from a psychological and educational perspective because they provide meaningful information on which features are more strongly

associated with each affective and behavioral construct of student engagement. Conversely, deep learning models are typically more complex and the model parameters are difficult to analyze and interpret.

5.2 Limitations and Future Work

In this study, the affect and behavior detectors were built for sixth-grade students in an urban public school as they learned topics from the subject matter of science in Betty's Brain. Sample size was limited due to the difficulties of observing affect and behaviors in class; researchers using expert labels to build detectors of student affect and behavior are unlikely to ever reach the millions of instances frequently employed in other deep learning domains. Thus, results should not be considered a reflection on the potential of deep learning in general but do suggest that it is unlikely to provide breakthroughs for student modeling applications where data is naturally limited.

Future research should explore the generalizability of our findings. For example, recent work indicated that affect detectors might not generalize well to new populations such as rural school students [29], and that deep learning models for affect and behavior detection were less effective in rural settings [20]. As such, it is especially meaningful to implement both approaches for different populations and test whether our comparison results generalize to other student populations, including rural and suburban students. Furthermore, will these results transfer to other contexts (e.g., other types of computer-based learning environments outside the domain of science)? Will the level of accuracy be comparable when we apply the two approaches to predict other constructs beyond affect detection?

Our findings have implications for the implementation of affect/behavior detectors to trigger interventions and observations in the open-ended learning environment. The advantages and disadvantages of each method should be considered in order to make decisions on which approach to pursue to detect affect/behavior with higher confidence in real time and to drive interventions. If the data set size is much larger than seen here, deep neural networks may provide the best results, but for our current data set size the choice of method appears to be based on the eventual application of the model: feature-engineered models for intervention, deep neural networks for discovery with models.

Acknowledgments. We would like to thank the National Science Foundation (NSF) for their support (#DRL-1561567).

References

1. Clarke-Midura, J., Yudelson, M.V.: Towards Identifying Students' Causal Reasoning Using Machine Learning. In: Proceedings of the 16th International Conference on Artificial Intelligence in Education (AIED 2013), pp. 704-707. Springer, Berlin, Heidelberg (2013)
2. Rowe, E., Asbell-Clarke, J., Baker, R.S., Eagle, M., Hicks, A.G., Barnes, T.M., Brown, R.A., Edwards, T.: Assessing Implicit Science Learning in Digital Games. Computers in Human Behavior. 76, 617-630 (2017)

3. Baker, R.S., Gowda, S.M., Wixon, M., Kalka, J., Wagner, A.Z., Salvi, A., Aleven, V., Kusbit, G.W., Ocumpaugh, J., Rossi, L.: Towards Sensor-Free Affect Detection in Cognitive Tutor Algebra. In: Proceedings of the 5th International Conference on Educational Data Mining, pp. 126-133 (2012)
4. D'Mello, S., Jackson, T., Craig, S., Morgan, B., Chipman, P., White, H., Person, N., Kort, B., Kaliouby, R.e., Picard, R., Graesser, A.: AutoTutor Detects and Responds to Learners Affective and Cognitive States. In: Proceedings of the Workshop on Emotional and Cognitive issues in ITS in conjunction with the 9th International Conference on ITS, pp. 31–43 (2008)
5. Pardos, Z.A., Baker, R.S., Pedro, M.O.C.Z.S., Gowda, S.M., Gowda, S.M.: Affective States and State Tests: Investigating How Affect Throughout the School Year Predicts End of Year Learning Outcomes. In: Proceedings of the 3rd International Conference on Learning Analytics and Knowledge, pp. 117-124 (2013)
6. Arroyo, I., Cooper, D.G., Burleson, W., Woolf, B.P., Muldner, K., Christopherson, R.: Emotion Sensors Go To School. In: Proceedings of the 2009 conference on Artificial Intelligence in Education (AIED 2009), pp. 17-24. IOS Press, Amsterdam, The Netherlands (2009)
7. Baker, R.S., Corbett, A.T., Roll, I., Koedinger, K.R.: Developing a Generalizable Detector of When Students Game the System. User Modeling and User-Adapted Interaction. 18, 287-314 (2008)
8. Cetintas, S., Si, L., Xin, Y.P., Hord, C.: Automatic Detection of Off-Task Behaviors in Intelligent Tutoring Systems with Machine Learning Techniques. IEEE Transactions on Learning Technologies. 3, 228-236 (2010)
9. Kai, S., Paquette, L., Baker, R.S., Bosch, N., D'Mello, S., Ocumpaugh, J., Shute, V., Ventura, M.: A Comparison of Video-based and Interaction-based Affect Detectors in Physics Playground. In: Proceedings of the 8th International Conference on Educational Data Mining, pp. 77-84 (2015)
10. Ocumpaugh, J., Baker, R.S., Rodrigo, M.M.T.: Baker Rodrigo Ocumpaugh Monitoring Protocol (BROMP) 2.0 Technical and Training Manual. Technical report, Teachers College, Columbia University, Ateneo Laboratory for the Learning Sciences (2015)
11. Fancsali, S.E.: Causal Discovery with Models: Behavior, Affect, and Learning in Cognitive Tutor Algebra. In: Proceedings of the 7th International Conference on Educational Data Mining (EDM 2014), pp. 28-35 (2014)
12. San Pedro, M.O.Z., Baker, R.S., Bowers, A.J., Heffernan, N.T.: Predicting College Enrollment from Student Interaction with an Intelligent Tutoring System in Middle School. In: Proceedings of the 6th International Conference on Educational Data Mining, pp. 177-184 (2013)
13. San Pedro, M.O.Z., Snow, E.L., Baker, R.S., McNamara, D.S., Heffernan, N.T.: Exploring Dynamic Assessments of Affect, Behavior, and Cognition and Math State Test Achievement. In: Proceedings of the 8th International Conference on Educational Data Mining, pp. 85-92 (2015)
14. LeCun, Y., Bengio, Y., Hinton, G.: Deep Learning. Nature. 521, 436–444 (2015)
15. Botelho, A.F., Baker, R.S., Heffernan, N.T.: Improving Sensor-Free Affect Detection Using Deep Learning. In: Proceedings of the 18th International Conference on Artificial Intelligence in Education (AIED 2017), pp. 40–51. Springer, Berlin, Heidelberg (2017)
16. Khajah, M., Lindsey, R.V., Mozer, M.C.: How Deep is Knowledge Tracing? In: Proceedings of the 9th International Conference on Educational Data Mining (EDM 2016), pp. 94–101 (2016)

17. Lin, C., Chi, M.: A Comparisons of BKT, RNN and LSTM for Learning Gain Prediction. In: Proceedings of the 18th International Conference on Artificial Intelligence in Education (AIED 2017), pp. 536–539. Springer (2017)
18. Piech, C., Spencer, J., Huang, J., Ganguli, S., Sahami, M., Guibas, L., Sohl-Dickstein, J.: Deep Knowledge Tracing. In: Advances in Neural Information Processing Systems 28 (NIPS 2015), pp. 505–513. Curran Associates, Inc. (2015)
19. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet Classification with Deep Convolutional Neural Networks. In: Proceedings of the 25th International Conference on Neural Information Processing Systems, pp. 1097–1105, Lake Tahoe, Nevada (2012)
20. Botelho, A.F., Baker, R.S., Heffernan, N.T.: Developing and Evaluating "Deep" Sensor-Free Detectors of Student Affect. (Manuscript in preparation)
21. Leelawong, K., Biswas, G.: Designing Learning by Teaching Agents: The Betty's Brain System. International Journal of Artificial Intelligence in Education. 18, 181-208 (2008)
22. Baker, R.S., Ocumpaugh, J., Gowda, S.M., Kamarainen, A.M., Metcalf, S.J.: Extending Log-Based Affect Detection to a Multi-User Virtual Environment for Science. In: Proceedings of the 22nd Conference on User Modelling, Adaptation, and Personalization, pp. 290-300 (2014)
23. Ocumpaugh, J., Baker, R.S., Rodrigo, M.M., Salvi, A., Velsen, M.v., Aghababyan, A., Martin, T.: HART: The Human Affect Recording Tool. In: Proceedings of the 33rd Annual International Conference on the Design of Communication (SIGDOC '15). ACM, New York, NY (2015)
24. Mierswa, I., Scholz, M., Klinkenberg, R., Wurst, M., Euler, T.: Rapid Prototyping for Complex Data Mining Tasks. In: Proc of KDD 2006, pp. 935-940 (2006)
25. Clevert, D.-A., Unterthiner, T., Hochreiter, S.: Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs). In: ICLR 2016, (2016)
26. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research. 15, 1929–1958 (2014)
27. Paquette, L., Baker, R.S., Pedro, M.A.S., Gobert, J.D., Rossi, L., Nakama, A., Kauffman-Rogoff, Z.: Sensor-Free Affect Detection for a Simulation-Based Science Inquiry Learning Environment. In: Proceedings of the 12th International Conference on Intelligent Tutoring Systems, pp. 1-10 (2014)
28. Sutskever, I., Vinyals, O., Le, Q.V.: Sequence to Sequence Learning with Neural Networks. In: Advances in Neural Information Processing Systems, pp. 3104–3112. Curran Associates, Inc. (2014)
29. Ocumpaugh, J., Baker, R.S., Gowda, S.M., Heffernan, N.T., Heffernan, C.: Population Validity for Educational Data Mining: A Case Study in Affect Detection. British Journal of Educational Psychology. 45, 487-501 (2014)