Towards Better Affect Detectors: Effect of Missing Skills, Class Features and Common Wrong Answers

Yutao Wang
Department of Computer Science
Worcester Polytechnic Institute
Massachusetts, USA 01609
yutaowang@wpi.edu

Neil T. Heffernan
Department of Computer Science
Worcester Polytechnic Institute
Massachusetts, USA 01609
nth@wpi.edu

Cristina Heffernan
Department of Computer Science
Worcester Polytechnic Institute
Massachusetts, USA 01609
ch@wpi.edu

ABSTRACT
The well-studied Baker et al., affect detectors on boredom, frustration, confusion and engagement concentration with ASSISTments dataset were used to predict state tests scores, college enrollment, and even whether a student majored in a STEM field. In this paper, we present three attempts to improve upon current affect detectors. The first attempt analyzed the effect of missing skill tags in the dataset to the accuracy of the affect detectors. The results show a small improvement after correctly tagging the missing skill values. The second attempt added four features related to student classes for feature selection. The third attempt added two features that described information about student common wrong answers for feature selection. Result showed that two out of the four detectors were improved by adding the new features.

Categories and Subject Descriptors
J.1 [**Administrative Data Processing**] Education; K.3.1 [**Computer Uses in Education**] Computer-assisted instruction (CAI)

General Terms
Measurement, Performance

Keywords
Measurement, Affect Detection, Missing Skill, Class Features, Common Wrong Answers, Learning Analytics

1. INTRODUCTION
Affect detection in educational systems is important in understanding student affect and its impacts on learning. Correctly detected student affect could potentially help guide interventions to improve student engagement, reduce student confusion, frustration or boredom. In recent years, sensor free affect detection (D'Mello et al. 2008, Baker et al. 2011, Sabourin et al. 2011) has gained more and more attention. This approach can be easily applied to various real-world educational systems for students' affec**t detection w**ithout requirement of sensor systems.

Currently, the best sensor free affect detectors were built by Baker et al. (2011) on the cognitive tutor dataset, which can be used to detect student engaged concentration, confusion, frustration and boredom solely from students' log data. The detectors were then rebuilt using the ASSISTments' dataset, and helped various of researches, including Hawkins et al.'s work (2013) on interface design influences affect. Pardos et al. (2013) investigated how affect influenced learning, and used affect conditions to predict state tests scores. San Pedro et al. (2013) even used the affect detector to analyze how affect influences the eventual decision to attend college, including college enrollments and whether a student majored in a STEM field.

The original sensor-free affect detection method has produced detectors that are better than chance, but not substantially better. However, in the three years since their creation, little has been reported on improvements to the original sensor-free affect detectors. In this paper, we made several attempts to improve the detectors. The first attempt was correcting the missing skill tags in the ASSISTments dataset. We found that the model was based upon the ASSISTments data that included almost a quarter of the questions not tagged with any skill. We decided to run experiments to see if tagging these questions with correct skills the detectors could perform better.

We also tried to improve the detectors by adding in new features. Two sets of features were considered. The first set was related to information about student classes. Student class is one of the most common objects that is studied in the educational field. However, when building student models such as models for predicting student performance or estimating student affective states, class level features are rarely considered. Wang et al. (2013) showed in a student model that class level parameters could be useful. In our experiments, results showed that class features also helped improve two out of the four affect detectors. The second set of features was related to whether the student made a common wrong answer. Previous research in our lab showed that the group of data logs in which all the answers are not common (namely uncommon wrong answer group) were more likely to be followed by a wrong attempt on the next problem for this particular student and skill. We looked at whether or not common wrong answers could also help improve affect detectors.

Details of all the models, including runnable versions in RapidMiner can be found online [7].

LAK '15, March 16 - 20, 2015, Poughkeepsie, NY, USA
Copyright 2015 ACM 978-1-4503-3417-4/15/03..$15.00
http://dx.doi.org/10.1145/2723576.2723618

2. METHODOLOGY

2.1 Dataset and features

The data used in the analysis presented here came from the ASSISTments system, a freely available web-based tutoring system for 4th through 10th grade mathematics. The system gives tutorial assistance if a student makes a wrong attempt or asks for help. Figure 1. shows an example of a hint, which is one type of assistance. A second type of assistance is presented if a student clicks on (or types in) an incorrect answer, at which point the student is given feedback that they answered incorrectly (sometimes, but by no means always, students will get a context-sensitive message we call a "buggy message").

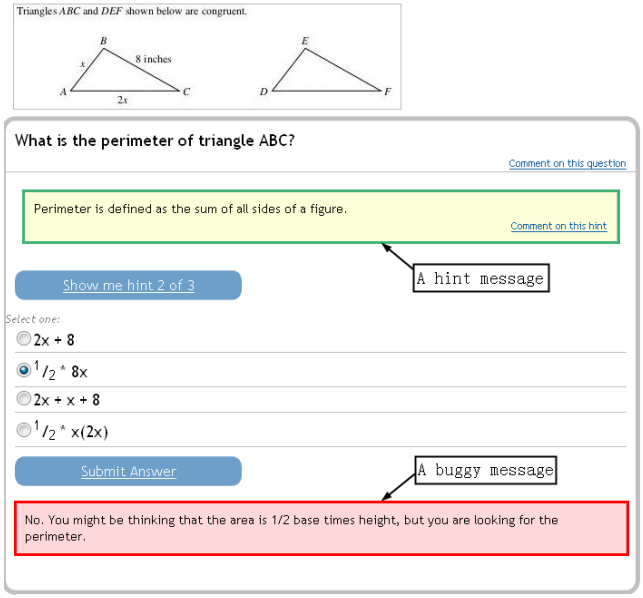

Figure 1. Hints and buggy message in ASSISTments

Our dataset also provides a special type of assistance called scaffolding as in Figure 2. Since it's only a small amount of our data, this detail might not be that important. But for completeness and for understanding of some of the features, we wanted to describe this. For those problems with scaffolding questions, if a student gets the original question wrong, the system will give the student a series of questions we call "scaffolding" that walk the student through the steps of solving the original question.

The students and features in this study were same as the previous studies of sensor free affect detectors on ASSISTments data (Ocumpaugh et al. 2014). Students were drawn from middle schools in the northeastern United States. The ground truth labels of student affect were obtained using quantitative field observations (QFOs) in the Baker-Rodrigo Observation Method Protocol (BROMP, Ocumpaugh et al. 2012). BROMP coders record the affective state of each student in a 20-second field observation window, which was later synchronized with student's interactions with the educational system according to the same internet time server to distill features for affect detection. Fifty-eight features, including temporal features, skill-based features, features based on the number of errors, the number of correct answers and the number of hints requested, were developed using the action data during and prior to the twenty seconds prior to data entry by the observer. Then mean, min, max and sum aggregators were used on these 58 features across the actions within the clip to generate a total of 232 features. Examples of features can be found in Tables 2 and 4.

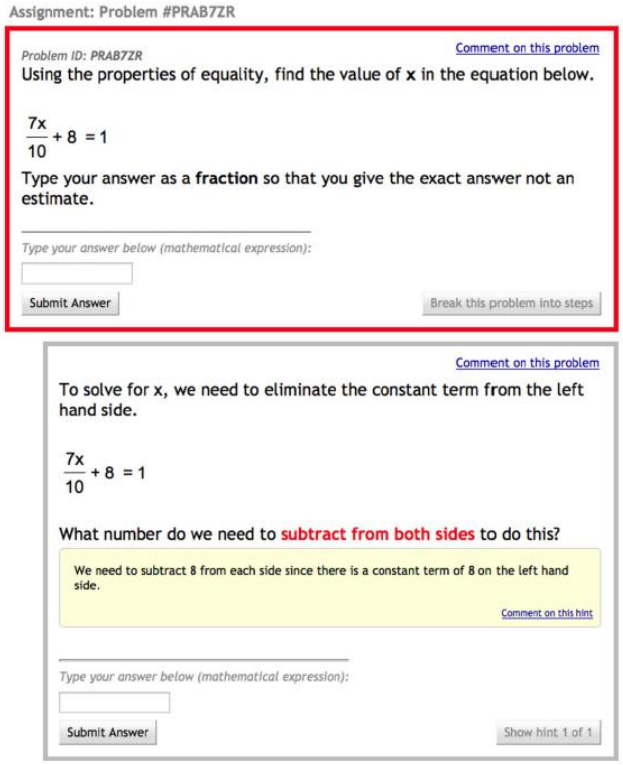

Figure 2. Scaffolding in ASSISTments

2.2 Classification and evaluation

The classification and evaluation method was described in detail in Baker et al. 2011. Four affect states: boredom, engaging concentration, frustration and confusion, were predicted separately by applying standard data mining classification algorithms within RapidMiner 5 (Mierswa et al., 2006). This resulted in four detectors, one for boredom, confusion, engaged concentration, and frustration respectively. The data mining algorithms selected include linear regression, decision trees, step regression, Naïve Bayes, JRip, J48, REPTree, Bayesian logistic regression, and K*. Forward selection feature selection was conducted for each of the machine learning algorithms using cross-validated kappa as the goodness metric.

Each detector was evaluated using five-fold student-level cross-validation to insure accuracy for new students. Students were split randomly into five groups. There were five rounds of training and testing. During each round a different group of students data served as the test set and the data from the remaining four groups served as the training set. Cohen's Kappa (Cohen 1960) and A' (Hanley and McNeil 1982) were used to determine which detectors were most effective. A Kappa of 0 indicates that the detector performs at chance; a Kappa of 1 indicates that the detector performs perfectly; and a Kappa of 0.2 indicates that the detector is 20% better than chance. A' is the probability that the algorithm will correctly identify whether a specific affective state is present or absent in a specific clip. A model with an A' of 0.5 performs at chance.

2.3 Missing skill problem

One of the biggest problems in "big-data" analysis is the missing data problem. In our dataset, we noticed that around 24% of the data has missing skill tags. All the logs that had missing skill tags were treated as a single skill: "no-skill." Since skill is one of the most important features in the educational dataset, we were intrigued to see how much improvement could be achieved if all the skills were tagged properly. To do so, we exported all 388 problems that had no skill tags in our dataset and manually tagged them with the correct skills. We then regenerated the 232 features using the new dataset and rebuilt all of the four affect detectors. The goal is to find out how much improvement, if any, can be achieved by generating more accurate skill related features.

2.4 Class and Common Wrong Answer features

Class features were generated based on the intuition that student affect could be influenced by the behavior of the class that they belonged to. For example, a student could feel less frustrated when he/she was better than most of other students in his/her class. We developed four new features that were related to student class:

- *pCorrectClass*: the percentage of correctness of all previous questions answered in this class;
- *pCorrectStudentPercentileRank*: the student percentile rank of average performance in this class so far for this student;
- *nClassData*: number of previous data points in this class;
- *nClassStudent*: number of students has been seen in this class so far;

These four features can be separated into two groups: *pCorrectClass* was designed to describe the average performance of the class that the student belonged. This could potentially be useful for normalizing the effect of student performance on student affect. *nClassData* was designed to evaluate the robustness of the feature *pCorrectClass*. When the number of data points in this class were small, we should put less trust in the *pCorrectClass* feature.

The *pCorrectStudentPercentileRank* feature was designed to describe how good the student did compared to his/her peer classmates. *nClassStudent* and *nClassData* could be used together to indicate how much we could trust the feature *pCorrectStudentPercentileRank*.

Common wrong answer is a novel feature that has great potential in educational models. Intuitively, students who answered a common wrong answer could indicate certain misunderstanding and/or understanding of the problem. We developed two features related to common wrong answers:

- *answerPercentage*: the percentage of this particular answer among all logs that answered this problem;
- *commonWrongAnswer*: a binary feature describes whether or not this answer is a common wrong answer; A common wrong answer was defined by answers that were given by at least 10% of the students that got the question wrong;

We evaluated the effect of class features and common wrong answer features using the same method. First, the four class features or the two common wrong answer features were added into the original 232 features. Then a unique-id was used to generate exactly the same resampling and cross-validation dataset to make sure the new detectors' performance can be directly compared with the original detectors. Finally, the same classification and evaluation methods were used to build and evaluate the new detectors.

3. EXPERIMENTAL RESULTS

3.1 Effect of missing skills

The results of correcting missing skills are shown in Table 1. The bold values in Table 1 showed the improved model results. As shown in the table, the improvement was small (only improved 3% of average Kappa). This indicated that the sensor free affect detectors could be safely used on datasets with a certain amount of missing skill tags, with only a small sacrifice of performance. This was good news for educational systems in which missing skill tags were inevitable (e.g., systems allowing teachers create their own problems without skill tags).

Table 1. Effect of missing skills

Detector	Original		With corrected skills	
	A'	Kappa	A'	Kappa
Engaged Concentration	0.731	0.417	**0.736**	**0.419**
Confusion	0.625	0.146	**0.627**	**0.148**
Frustration	0.597	0.151	**0.602**	**0.157**
Boredom	0.662	0.243	**0.671**	**0.264**
Average	0.654	0.239	**0.659**	**0.247**

As an example, features automatically selected for the engaged concentration detector were listed in Table 2. At the first glance, the two sets of features seemed different. However, by closely looking at what the features represent, many of them were providing similar information.

Table 2. The features in the final detectors after correcting missing skills

Engaged Concentration	
Original	With Corrected Skills
Total first responses attempted in the tutor so far.	The sum of numbers of first responses during school hours (between 7:00 am and 3:00 pm)
The number of main problems seen in this 20 seconds	The sum of time spend on this problem
The minimal number of first responses during school hours (between 7:00 am and 3:00 pm)	The average correctness in this 20 seconds
The average time spent on first responses in answering scaffolding problems	The maximum number of first responses that were help requests
The average correctness in this 20 seconds	The average number of total hints

The maximum number of previous incorrect actions and help requests for any skill in the clip	

3.2 Effect of class and Common Wrong Answer features

After adding four class features, detector performance was improved in two out of four detectors. Results are shown in Table 3. The bold values showed the improved model results. For confusion and frustration detectors, the class features were not selected into the final models.

Table 3. Effect of class features

Detector	Original		With Class Features	
	A'	Kappa	A'	Kappa
Engaged Concentration	0.731	0.417	**0.743**	**0.423**
Confusion	0.625	0.146	0.625	0.146
Frustration	0.597	0.151	0.597	0.151
Boredom	0.662	0.243	**0.671**	**0.260**
Average	0.654	0.239	**0.659**	**0.245**

Table 4. The features in the final detectors with class features

Engaged Concentration	
Original	**With Class Features**
Total first responses attempted in the tutor so far	**The percentage of correctness of all the questions answered in the class so far**
The number of main problems seen in this 20 seconds	**Number of data points for this class so far**
The minimal number of first responses during school hours (between 7:00 am and 3:00 pm)	The number of main problems seen in this 20 seconds
The average time spent on first responses in answering scaffolding problems	The average of the number of first responses during school hours (between 7:00 am and 3:00 pm)
The average correctness in this 20 seconds	
The maximum number of previous incorrect actions and help requests for any skill in the clip	

Boredom	
Original	**With Class Features**
The sum of the number of first responses during school hours (between 7:00 am and 3:00 pm)	The sum of the number of first responses during school hours (between 7:00 am and 3:00 pm)
Sum of wrong answers in the past 8 problems	**The number of students has been seen in this class so far**
The average of response times for any skill in the clip	**The student percentile rank of average performance in the class so far for this student**
The average of the number of first responses during school hours (between 7:00 am and 3:00 pm)	The minimal number of multiple choice questions in this 20 seconds
Sum of wrong answers in the past 5 problems	The minimal number of hints in this 20 seconds
	The minimal number of questions that has a help request as the first response

For the two improved detectors: engaged concentration and boredom, features automatically selected for each of the detectors during machine learning are listed in Table 4.

In the engaged concentration detector with class features, the percentage of correctness of the class (*pCorrectClass*) and the number of data points in the class (*nClassData*) were selected. This could indicate that in modeling concentration, class performance is more important than student individual performance.

In the boredom detector with class features, the student percentile rank of average performance in the class (*pCorrectStudentPercentileRank*) and number of students (*nClassStudent*) were selected. These features replaced the performance feature that described how many incorrect answers the student answered before as in the original boredom detector, while bring in features describing how many help students can get from the system, including whether or not the question is multiple choice question, and how many hints were asked. The result suggests that for boredom detector, student performance can be more effectively represented by students' percentage rank in correctness.

After adding two common wrong answer features, detector performance was recalculated. The binary version of common wrong answer feature was selected into the engaged concentration detector: students who give common wrong answers are more likely to be effectively working. The more detailed version of common wrong answer features was selected into the boredom detector. Certain, but not all, common wrong answers are related to guessing, which is a common behavior of bored students.

For confusion and frustration, the common wrong answer features were not selected into the final models.

This approach achieved 5% improvement on average Kappa of the detectors. Result tables can be found online [7]

4. DISCUSSION AND CONCLUSIONS

In this paper, we presented three attempts to improve existing sensor-free affect detectors with the ASSISTments dataset. The first attempt analyzed the effect of missing skill tags in the dataset on the accuracy of the affect detectors. Not many researchers pay attention to the performance of models in dataset with missing values in the learning analytic field. The results showed only a small improvement after correctly tagging the missing skill values.

This suggests that it should be safe to use the affect detectors with a certain amount of missing skills.

The second attempt added four features that describe information about student classes into the feature pool for feature selection. Class is one of the common objects that are studied in learning analytics analyses. Results showed that class features helped improve the concentration and the boredom detectors by 3.5% on average Kappa.

The third attempt added two features that describe information about how common the student's answer was. This approach achieved 5% improvement on average Kappa. The result showed that there is potential in this novel feature in learning analytics analyses. Many public dataset available online (e.g. the PSLC DataShop) decided to not reveal student answers out of a concern for privacy: in theory, someone could type in something that might give out their identity (e.g., "I am Barrack Obama and I don't know how to do this problem!"). Of the 10 million answers a year we get at ASSISTments, a small portion might have such information. We suggest a compromised approach for sharing student answer data: release what the student types in for all common wrong answers using our operationalization of common wrong answer. If the student answered something unique, then it would not be shared.

The result in this paper is likely to generalize to new systems which give hints. Many systems, such as cognitive tutor, Mastering Physics can easily compute most of the features in our models. For the class level features you do need to know the concept of class. Since most homework support systems have students nested inside of classes, the result should apply to many others systems.

This work is still at the early stages. We see it as one of the incremental steps to build a useful tool for understanding and automatically adapting to differences in learner affect. There is still substantial room for improvement in compare with expert coders' Kappa values (around 0.6 or 0.7). More features and different methods could be used to further improve the detectors. In the long-term, we could incorporate these detectors into the ASSISTments platform to help teachers to understand students' affective states or provide interventions aim for better learning outcomes.

5. ACKNOWLEDGMENTS

The first author is funded under NSF grant 1252297. We appreciate funding from NSF (# 1440753, 1316736, 1252297, 1109483, 1031398, 0742503), ONR's 'STEM Grand Challenges' and IES (# R305A120125 & R305C100024). We also want to thank Bohao Li for the suggestion to look at the common wrong answers.

6. REFERENCES

[1] Baker, R.S.J.d., Gowda, S., Corbett, A.T. 2011. Towards predicting future transfer of learning. *Proceedings of 15th International Conference on Artificial Intelligence in Education*, 23-30.

[2] Cohen, J. A 1960. Coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1), 37-46.

[3] Craig, S. D., Graesser, A. C., Sullins, J. and Gholson, B. 2004. Affect and learning: an exploratory look into the role of affect in learning with AutoTutor. *Journal of Educational Media*, 29(3), 241–250.

[4] D'Mello, S.K., Craig, S.D., Witherspoon, A. W., McDaniel, B. T., and Graesser, A. C. 2008. Automatic Detection of Learner's Affect from Conversational Cues. *User Modeling and User- Adapted Interaction*, 18(1-2), 45-80.

[5] Hawkins, W., Heffernan, N. and Baker, R. S. J. d. 2013.Which is more responsible for boredom in intelligent tutoring systems: students (trait) or problems (state)? *Proceedings of the 5th biannual Conference on Affective Computing and Intelligent Interaction*, 618–623.

[6] Hanley, J., and McNeil, B. 1982. The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. *Radiology*, 143, 29-36.

[7] Heffernan, N. T. 2014. ASSISTments Data. Accessed on January 30, 2015, from: https://sites.google.com/site/assistmentsdata/home/

[8] Mierswa, I.,Wurst, M., Klinkenberg, R., Scholz, M. and Euler, T. 2006. YALE: rapid prototyping for complex data mining tasks. *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 935–940.

[9] Ocumpaugh, J., Baker, R. S. and Rodrigo, M.M. A. 2012. *Baker-Rodrigo Observation Method Protocol (BROMP) 1.0. Training Manual version 1.0*. Technical Report. New York, NY: EdLab. Manila, Philippines: Ateneo Laboratory for the Learning Sciences.

[10] Ocumpaugh, J., Baker, R., Gowda, S., Heffernan, N., and Heffernan, C. 2014. Population validity for educational data Mining models: A case study in affect detection. *British Journal of Educational Technology*, 45(3), 487–501.

[11] Pardos, Z. A., Baker, R. S. J. d., San Pedro,M. O. C. Z., Gowda,S. M. and Gowda, S. M. 2013. Affective states and state tests: investigating how affect throughout the school year predicts end of year learning outcomes. In *Proc. of the Third International Conference on Learning Analytics and Knowledge*, ACM, New York, USA, 117-124.

[12] Sabourin, J., Mott, B., and Lester, J. 2011. Modeling Learner Affect with Theoretically Grounded Dynamic Bayesian Networks. *Proceedings of the 4th International Conference on Affective Computing and Intelligent Interaction*, 286-295.

[13] San Pedro, M. O. Z., Baker, R. S. J. d., Bowers, A. J. and Heffernan, N. T. 2013. Predicting college enrollment from student interaction with an Intelligent Tutoring System in middle school. *Proceedings of the 6th International Conference on Educational Data Mining*, 177–184.

[14] Wang, Y., Beck, J.E. 2013. Class vs. Student in a Bayesian Network Student Model. *Proceedings of 16th International Conference on Artificial Intelligence in Education*. 151-160.