Trivedi S., Pardos Z. A., Heffernan N. T., In Press (2011) Clustering Students to Generate an Ensemble to Improve Standard Test Score Predictions. In *Proceedings of the 15th International Conference on Artificial Intelligence in Education*, Auckland, New Zealand.

Clustering Students to Generate an Ensemble to Improve Standard Test Score Predictions

Shubhendu Trivedi, Zachary A. Pardos, Neil T. Heffernan

Department of Computer Science, Worcester Polytechnic Institute,
Worcester, MA-01609 United States
{s_trivedi, zpardos, nth}@wpi.edu

Abstract. In typical assessment student are not given feedback, as it is harder to predict student knowledge if it is changing during testing. Intelligent Tutoring systems, that offer assistance while the student is participating, offer a clear benefit of assisting students, but how well can they assess students? What is the trade off in terms of assessment accuracy if we allow student to be assisted on an exam. In a prior study, we showed the assistance with assessments quality to be equal. In this work, we introduce a more sophisticated method by which we can ensemble together multiple models based upon clustering students. We show that in fact, the assessment quality as determined by the assistance data is a better estimator of student knowledge. The implications of this study suggest that by using computer tutors for assessment, we can save much instructional time that is currently used for just assessment.

Keywords: Clustering, Ensemble Learning, Intelligent Tutoring Systems, Regression, Dynamic Assessment, Educational Data Mining

1 Introduction

Feng *et al.*[1] reported the counter-intuitive result that data from an intelligent tutoring system could better predict state test scores if it considered the extra measures collected while providing the students with feedback and help. These measures included metrics such as number of hints that students needed to solve a problem correctly and the time it took them to solve. That paper [1] was judged as best article of the year at User Modeling and User-Adapted Interaction and was cited in the National Educational Technology plan. It mentions a weakness of the paper concerning the fact that time was never held constant. Feng *et al.* go one step ahead and controlled for time in following work [2]. In that paper, students did half the number of problems in a dynamic test setting (where help was administered by the tutor) as opposed to the static condition (where students received no help) and reported better predictions on the state test by the dynamic condition, but the difference was not statistically reliable. This present work starts from Feng *et al.* [2] and investigates if the dynamic assessment data can be better utilized to increase prediction accuracy over the static condition. We use a newly introduced method that clusters students, creates a mixture of experts and then ensembles the predictions made by each cluster model to achieve a reliable improvement.

2 Literature Review

The Bayesian knowledge tracing model [3] and its variants [4] [5] have become the mainstay in the Intelligent Tutoring System (ITS) community to track student knowledge. This knowledge estimate is used for calibrating the amount of training students require for skill mastery. One of the most important aspects of such modeling is to ensure that performance on a tutoring system is transferred to actual post tests. If this is not the case, then that implies over-training within the tutoring system. In fact, it is reasonable to say that one of the most important measures of success of a tutoring system is its ability to predict student performance on a post-test. Since such a transfer is dependent on the quality of assessment, a tension exists between focusing on quality of assessment and quality of student assistance.

Traditionally, performance on a post-test is predicted by using practice tests. Practice tests based on past questions from specific state tests can give a crude estimate of how well the student might perform in the actual state test. Improving this estimate would be highly beneficial for educators and students. For improving such assessment, dynamic assessment [6] has long been advocated as an effective method. Dynamic assessment is an interactive approach to student assessment that is based on how much help a student requires during a practice test. Campione *et al.* [7] compared the traditional testing paradigm, in which the students are not given any help, with a dynamic testing paradigm in which students are given graduated hints for questions that they answer incorrectly. They tried to measure learning gains for both the paradigms from pre-test to post-test and suggested that such dynamic testing could be done effectively with computers. Such assessment makes intuitive sense as standard practice tests simply measure the percent of questions that a student gets correct. This might not give a good estimate of a student's knowledge limitations. If a student gets a question wrong, it might not necessarily imply absence of knowledge pertaining to the question. It is likely that the student has some knowledge related to the question but not enough to get it correct. It is thus desirable to have a fine grained measure of the knowledge limitations of the student during assessment. Such a measure might be obtained by monitoring the amount of help the student needs to get to a correct response from an incorrect response. ITS provide the tools for doing dynamic assessment more effectively as they adapt while interacting with individual students and make it easier to provide interventions and measure their effect. Fuchs *et al.* [9] studied dynamic assessment focusing on unique information, such as how responsive a user is to intervention. Feng *et al.* [1][2] used extensive information collected by the ASSISTments tutor [13] to show that the dynamic assessment gives a relatively better prediction as compared to static assessment. This work effectively showed that dynamic assessment led to better predictions on the post test. This was done by fitting a linear regression model on the dynamic assessment features and making predictions on the MCAS test scores.

They concluded that while dynamic assessment gave good assessment of students, the MCAS predictions made using those features lead to only a marginally statistically significant improvement as compared to the static condition. In this paper we explored the dynamic assessment data to see if we could make significantly better predictions on the MCAS test score. A significant result would further validate the use of ITS as a replacement to static assessments.

2 Data

The dataset that we considered was the same as used by Feng *et al.*[2]. It comes from the 2004-05 school year, the first full year when ASSISTments was used in two schools in Massachusetts. ASSISTments is an e-learning and e-assessing research platform [10] developed at Worcester Polytechnic Institute. Complete data for the 2004-05 year was obtained for 628 students. The data contained the dynamic interaction measures of the students and the final grades obtained in the state test (MCAS) taken in 2005. The dynamic measures were aggregated as students used the tutor.

2.1 Metrics

The following metrics were developed for dynamic testing by Feng *et al.* [2] and were used in these experiments. They try to incorporate a variety of features that summarize a student's performance in the system. The features were as follows: 1) the student's percent correct on the main problems 2) number of problems done 3) percent correct on the help questions 4) average time spent per item 5) average number of attempts per item and 6) average numbers of hints per item. Out of these, only the first was as a static metric and was used to predict the MCAS score in the static condition. The other five and a dynamic version of student's percent correct on the main problems were used to make predictions in the dynamic condition.

The predictions were made on the MCAS scores. The MCAS or the Massachusetts Comprehensive Assessment System is a state administered test. It produces tests for English, Mathematics, Science and Social Studies for grades 3 to 10. The data set we explore is from an 8^{th} grade mathematics test.

3 Methodology

The data was split into randomly selected disjoint 70% train and 30% test sets. Feng *et al.*[2] fit a stepwise linear regression model using the dynamic assessment features on the training set to make a prediction on the MCAS scores on the test set. They reported an improvement in prediction accuracy with a marginal statistical significance relative to the predictions made only using data from the static condition. Fitting in a single linear regression model for the entire student data might be a bad idea for two reasons. First, the relationship between the independent variables (dynamic assessment features) and the dependent variables (MCAS test scores) might not be a linear one. If so, training a linear model would have high bias for the data and no matter how much data is used to train the model, there would always be a high prediction error. The second conceivable source of error is related to the first. A student population would have students with varying knowledge levels, thus requiring different amounts of assistance. Thus it might be a bad idea to fit the entire population in a single model. Students often fall into groups having similar knowledge levels, assistance requirements, etc. It is thus worth attempting to fit different models for

different groups of students. It, however, must be noted that while such groups could be identified using clustering, the groups obtained may not be easily interpretable.

3.1 Clustering

The previous section mentions that it might not be a good idea to fit in a single model for the entire student population and that there might exist groups of students having similar knowledge levels and nature of responses to interventions. A natural method to find such patterns in the data is by clustering. If data was generated by a finite set of distinct processes, then clustering methods are maximum likelihood methods to identify such underlying processes and separating them. The idea in this work is to fit in a linear regression model for each such group in the training set. The prediction for the MCAS score for each student from the test set would thus involve two steps: identification of the cluster to which the student from the test set belongs and then using the model for that cluster to make the prediction of the MCAS score for the student.

We used K-means clustering for the identification of K groups. The initialization of cluster centroids was done randomly and the clusters were identified by using Euclidean distance. K-means finds out the best separated clusters by trying to minimize a distortion function. The distortion function is a non-convex function and thus implies that K-means is susceptible to getting stuck in local optima. This means that when K-means is run with random cluster centroids; we might not reach the best solution possible. To reduce the chances of getting a sub-optimal clustering we restarted K-means 200 times with random initialization.

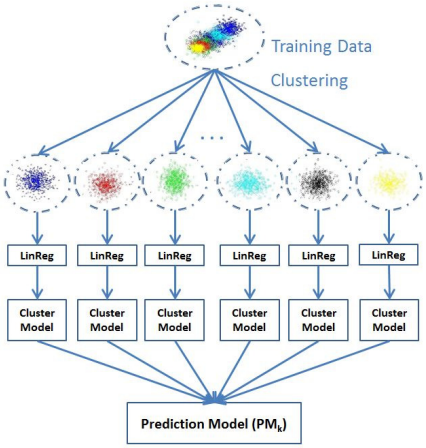

Fig.1. Schematic illustrating the steps for obtaining a prediction model (PM_K). There would be one such prediction model for each value of K chosen (1 to K would give K prediction models).

For each cluster identified we trained a separate linear regression model (Fig. 1). We call such a linear regression model (for each cluster) a cluster model. For data separated into K clusters there would be K cluster models. All of these K cluster models taken together make predictions on the entire test set. These K cluster models

together can be thought to form a more complex model. We call such a model a prediction model i.e. PM_K, with the subscript K identifying the number of cluster models in the prediction model. Feng *et al.* [2] used the prediction model PM_1, since only a single linear regression model was fit over the entire data-set. The value of K can be varied from 1 to K to obtain K prediction models. For example: if K = 1, 2 and 3, there would be three prediction models - PM_1 having a single cluster model (K=1), PM_2 having two different cluster models (K=2) and PM_3, that is the prediction model with three different cluster models (K=3). It is noteworthy that the cluster models in different prediction models would be different.

If K prediction models are constructed from the data, there would be a set of K different predictions on the test data. These predictions are compared with those obtained on PM_1, i.e. a linear regression model fit over the entire data-set to see if there is an improvement in prediction accuracy. An improvement would indicate a strong result that dynamic assessment indeed gives a much better assessment of student learning.

3.2 Ensemble Learning

Section 3.1 described how, by using K as a controllable parameter, we can obtain a set of K prediction models and K corresponding predictions. The training data is first clustered by K-means and K clusters are obtained. For each of the clusters we fit a linear regression model, which we called the cluster model. The cluster models together are referred to as a prediction model. This prediction model makes a prediction on the entire test set. But since K is a free parameter, for each value of K we get a different prediction model and a different set of predictions. For example when K=2, the prediction model will have two cluster models. When K=7, the prediction model will have 7 cluster models. Thus, by means of clustering, we generate a number of prediction models.

While we are interested in looking at how each prediction model performs. It would also be interesting to look at ways in which the K predictions can be combined together to give a single prediction. Such a combination of predictors leads to ensembling. Ensemble methods have seen a rapid growth in the past decade in the machine learning community [12][13][14].

An ensemble is a group of predictors each of which gives an estimate of a target variable. Ensembling is a way to combine these predictions with the hope that the generalization error of the combination is lesser than each of the individual predictors. The success of ensembling lies in the ability to exploit diversity in the individual predictors. That is, if the individual predictors exhibit different patterns of generalization, then the strengths of each of the predictors can be combined to form a single stronger predictor. Dietterich [12] suggests three comprehensive reasons why ensembles perform better than the individual predictors. Much research in ensembling has gone into finding methods that encourage diversity in the predictors.

3.2.1 Methodology for Combining the Predictions

We have a set of K predictors. The most obvious way of combining them is by some type of averaging. The combination could also be done using Random Forests [15], but they have not been explored in this work as we are extending work that simply used linear regression. We explored two methods for combining these predictors.

1. Uniform Averaging: This is the simplest method for combining predictions. The K predictions obtained (as discussed in section 3.1) are simply averaged to get a combined prediction. In addition to averaging all predictions we could also choose to average just a subset of the predictions together.

2. Weighted averaging: In uniform averaging, each predictor is given the same weight. However, it is possible that the predictions made by some model are more important than the predictions made by another model. Thus, it is reasonable to combine the models by means of a weighted average. Such weighted averaging could be done by means of a linear regression. Since we did not find an improvement with weighted averaging, the methodology and results are not discussed in detail.

4 Results

4.1 Prediction Models

The data was first clustered with K taken from 2 to 7. Clustering beyond 7 clusters was problematic as it returned empty clusters. Hence the experiments were restricted to a maximum of K=7 for this dataset. The prediction on the MCAS was made first by using PM_1. Then, K was varied from 2 to 7 and a set of six more predictions on the MCAS were obtained (all dynamic features were used). The Mean Absolute Difference (MAD) and the Root Mean Square Errors (RMSE) of the MCAS in the test set were found. This section summarizes these results. It also compares the results with the static condition.

Table 1. Prediction errors by different prediction models

Model	**MAD**	**p-value (with PM_1)**	**p-value (with static)**	**RMSE**
Static	10.4900	0.0180	-	12.7161
PM_1	9.6170	-	0.0180	11.5135
PM_2	9.3530	0.1255	0.0036	11.4286
PM_3	9.3522	0.2005	0.0074	11.4377
PM_4	9.3005	0.1975	0.0062	11.5243
PM_5	9.3667	0.3375	0.0067	11.7291
PM_6	9.3518	0.2347	0.0052	11.5100
PM_7	9.4818	0.6138	0.0134	11.6762

Almost all Prediction Models (Table 1) showed a statistically significant improvement in prediction as compared to the static condition demonstrating greater

assessment power using the dynamic condition. However, though there is an improvement in the error as compared to the Prediction Model 1, the improvement is not statistically significant, as was previously found to be the case [1].

4.2 Averaging Predictions

As reported in section 4.1 the prediction models do not show a statistically significant improvement in prediction accuracy of the MCAS score relative to the PM_1. As discussed in section 3.2, combining them might lead to improved predictions. This section reports these results.

Table 2. Prediction errors by different prediction models averaged. The subscripts refer to the models whose predictions were used in averaging.

Model	**MAD**	**p-value (with PM_1)**	**p-value (with static)**	**RMSE**
Static	10.4900	0.0180	-	12.7161
PM_1	9.6170	-	0.0180	11.5135
$PM_{1 \text{ to } 4}$	9.2375	0.0192	0.0013	11.3042
$PM_{1 \text{ to } 5}$	9.2286	0.0251	0.0012	11.3405
$PM_{1 \text{ to } 6}$	9.2268	0.0260	0.0012	11.3412
$PM_{1 \text{ to } 7}$	9.2398	0.0365	0.0013	11.3511
$PM_{2 \text{ to } 4}$	9.2604	0.0526	0.0022	11.3379
$PM_{2 \text{ to } 5}$	9.2406	0.0540	0.0018	11.3818
$PM_{2 \text{ to } 6}$	9.2348	0.0475	0.0016	11.3753
$PM_{2 \text{ to } 7}$	9.2507	0.0630	0.0017	11.3830

Averaging across prediction models clearly improves predictions as compared to the prediction models taken alone (Table 2). The improvement is not just in the error but also in terms of statistical significance and thus improves the results reported in 4.1. These results validate the idea that clustering helps in predictions. These results show how the dynamic assessment prediction accuracy can be further improved.

4 Contributions

This paper makes one clear contribution. This is the first paper we know of that clearly demonstrates that not only can an Intelligent Tutoring System allow students to learn while being assessed but also indicates a significant gain in assessment accuracy. This is important, as many classrooms take away time from instruction to administer tests. If we can provide such a technology it would save instruction time and give better assessment and would thus be highly beneficial to students and instructors. The second contribution of this paper is the application of clustering student data and ensembling predictions that we are introducing to the field in a KDD paper [16]. In that paper we applied this approach to a number of datasets from the

UC Irvine Machine Learning repository and reported a prediction improvement in all datasets.

References

1. Feng, M., Heffernan. N. T. & Koedinger. K. R. (2009). Addressing the assessment challenge in an online system that tutors as it assesses. *User Modeling and User-Adapted Interaction: The Journal of Personalization Research*. 19(3). 2009.
2. Feng. M., Heffernan. N. T., (2010). Can We Get Better Assessment From A Tutoring System Compared to Traditional Paper Testing? Can We Have Our Cake (better assessment) and Eat it too (student learning during the test). *Proceedings of the 3rd International Conference on Educational Data Mining*?, 41-50.
3. Corbett. A. T. & Anderson, J. R. (1995). Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User Modeling and User Adapted Interaction, 4, 253-278.*
4. Pardos, Z.A., Heffernan, N. T. In Press (2011) Using HMMs and bagged decision trees to leverage rich features of user and skill from an intelligent tutoring system dataset. *Journal of Machine Learning Research C & WP.*
5. Baker, R. S. J. d., Corbett. A. T., Aleven, V. (2008) More Accurate Student Modeling Through Contextual Estimation of Guess and Slip Probabilities in Bayesian Knowledge Tracing. *Proceedings of the 14th International Conference on Artificial Intelligence in Education.* Brightion, UK, 531-538
6. Grigerenko, E. L. & Steinberg, R. J. (1998). Dynamic Testing. *Psychological Bulletin, 124, 75-111*
7. Campione. J. C. & Brown. A. L. (1985). Dynamic Assessment: One Approach and some Initial Data. Technical Report. No. 361. Cambridge, MA. Illinois University, Urbana, Center for the Study of Reading. ED 269735
8. Fuchs. L. S., Compton. D. L. Fuchs. D., Hollenbeck. K. N., Craddock. C. F., & Hamlett. C. L. (2008). Dynamic Assessment of Algebraic Learning in Predicting Third Graders' of Mathematical Problem Solving. *Journal of Educational Psychology*, 100(4), 829-850.
9. Fuchs. D., Fuchs L.S., Compton. D. L., Bouton. B., Caffrey. E., & Hill. L., (2007). Dynamic Assessment as Responsiveness to Intervention. *Teaching Exceptional Children.* 39(5), 58-63.
10. Razzaq. L., Feng M., Nuzzo-Jones. G., Heffernan. N. T., Koedinger. K. R., Junker. B., Ritter. S., Knight A., Aniszczyk. C., Choksey. S., Livak. T., Mercado. E., Turner. T. E., Upalekar R., Walonoski. J.A., Macasek. M. A., & Rasmussen. K. P. (2005). The Assistment Project: Blending Assessment and Assisting. In C. K. Looi, G. McCalla, B. Bredeweg, & J. Breuker (Eds). *Proceedings of the 12th International Conference on Artificial Intelligence in Education, Amesterdam.* ISO Press, pp 555-562.
11. Baker, R. S., Corbett, A.T., Koedinger, K. R., & Wagner, A. Z. (2004). Off-task behaviour in the Cognitive Tutor Classroom: When Students "game the system". *Proceedings of the ACM CHI 2004: Computer - Human Interaction*. (pp. 383-390). New York: ACM.
12. Dietterich. T. G. (2000). Ensemble Methods in Machine Learning, *In First International workshop on Multiple Classifier Systems,* J. Kittler and F. Roli (Eds.), Lecture Notes in Computer Science, New York, Springer Varlag, 1-15.
13. Dietterich. T. G. (2000). An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization, *Machine Learning, Kluwer Academic Publishers*. Vol. 40, pp 139-157.
14. Brown. G., Wyatt. J. L., Tino. P., (2005). Managing Diversity in Regression Ensembles, *Journal of Machine Learning Research*. Vol 6, pp 1621-1650.
15. Breiman. L., (2001). Random Forests, *Machine Learning,* Vol 45(1), pp 5-32
16. Trivedi, S., Pardos, Z. A., Heffernan, N. T, (In Submission) The Utility of Clustering in Prediction Tasks. In submission to the 17th Conference on Knowledge Discovery and Data Mining, 2011.