PREDICTING STUDENT PERFORMANCE FOR AN INTELLIGENT TUTORING SYSTEM USING SUPPORT VECTOR MACHINE AND BAYESIAN KNOWLEDGE TRACING

Anjali Doneria ADONERI@NCSU.EDU

Smit Doshi STDOSHI@NCSU.EDU

Editor: Dr. Colin F. Lynch

1. Introduction

Nowadays, the use of Intelligent Tutoring Systems in academic setting is becoming quite common. Students get the help of tutors to solve exercises and do their assignments or learn lessons. These tutors are equipped with technology to suggest problems to the student based on their learning capabilities, help them through the steps of solving a problem, suggest steps if the student asks for help or give feedback as and when needed. The use of intelligent tutors enables collection of huge amount of data that can be mined to predict to say if a student will get a question correct or incorrect, or if a student is proficient in a certain skill or task or knowledge component. Thus, accurately predicting student performance based on their ongoing activities becomes crucial for effectively carrying out pedagogical interventions to ensure students satisfactory performance and help the instructors in knowing which student needs special attention in which area of the subject. Different data mining techniques have been applied to educational data to enhance teaching methods, improve the quality of teaching, identify weak students, and identify factors that influence a students academic performance. This helps in bringing the benefits to students, educators and academic institutions.

As a part of this project, we are presenting two prediction models to assess student performance for a given set of tasks while learning through an intelligent tutoring system. Support Vector Machines (SVM) are very popular and powerful prediction learning technique so we have made use of them to learn from the knowledge components and step duration for the correct attempt. Bayesian Knowledge Tracing (BKT) is a very popular algorithm used in intelligent tutoring systems to model each learners skill mastery. Hence we are running both the models on the KDDCup 2010 Dataset to analyse prediction results by SVM against BKT prediction.

We first converted all the nominal data to numeric data and got maximum and minimum for the fields. Next, we performed feature selection using step forward feature selection algorithm to extract fields affecting the results the most. Furthermore, we built student models using SVM linear kernel doing 5-cross validation and 80-20% data splitting for each student data. Linear SVM was shortlisted after comparing accuracies for linear, rbf and sigmoid kernel. We then developed BKT model using different probability equations. In the end, we compared the prediction results from both the algorithms and drawn analysis and differences about them. The results showed that SVM slightly outperformed BKT for predicting student performance.

2. Background

A lot of research has been done in the field of predicting student performance using different machine learning algorithms. Different machine learning techniques like decision trees[7], artificial neural networks[8], matrix factorization[9], collaborative filtering[10] have been extensively applied to develop prediction algorithms for student performance. Most of these works do not take into account the sequential effect of students improving their knowledge over the course of time, hence the prediction is treated as a one-time task. To take these sequential effects into account, a three-mode tensor factorization technique[11] has been developed to predict student performance for ITSs. This technique takes into account student, problem and time into consideration to build the model. Al-Shehri, Huda, et al. [12] have applied Support Vector Machines and K-Nearest Neighbours on the dataset provided by University of Minho in Portugal to predict final grade of students falling under the range of 0 and 20. They used linear SVM and in their analysis, SVM slightly outperformed K-Nearest Neighbours which showed a higher correlation coefficient of 0.96 and relative absolute error of 19.92%. Oloruntoba, S. A., and J. L. Akinode [13] also applied Support Vector Machines to analyse the relationship between students pre-admission academic profile and his final academic performance by using data from one of the Federal Polytechnic in south West part of Nigeria. They compared results of using RBF kernel with penalty (C=100) with other ML techniques such as KNN, Decision Trees and linear regression and obtained the highest training accuracy of 94% and prediction accuracy of 97% for SVM model.

We are using educational data mining competition dataset from 2010 KDD Cup which was hosted by ACM for predicting correctness of a given math step given information about the step and the students past history of responses. As part of the competition, the team from National Taiwan University (NTU) was the winner of the competition followed by the team from Worcester Polytechnic Institute. As Yu, Hsiang-Fu et al. [1] mentions, the team from NTU consisted of three instructors, two TAs, 19 students and one RA forming six different sub-teams. Most students teams expanded features by binarization and discretization techniques with the resulting sparse feature sets being trained by logistic regression. One subteam used condensed features using simple statistical techniques and applied random forest for training. They combined the results from the student subteams by regularized linear regression for their final submission. They were not able to obtain good results by direct modeling of domain knowledge using Bayesian network, so they used traditional classification technique of logistic regression. Zachary A. Pardos and Niel Heffernan [2] developed an ensemble method by combining customised hidden markov models with features extracted and Random forest-decision tree modeling. In their case, prediction error was very low for rows that had sufficient data to compile a complete user and skill feature set however the error was very high for rows where the user did not have sufficient skill data.

3. Methodology

3.1 Data Description

The Association for Computing Machinery (ACM) used an educational dataset for hosting a competition in 2010. The datasets for this 2010 Knowledge Discovery and Data Mining Cup (KDD Cup) came from Intelligent Tutoring Systems (ITS) used by thousands of students over the course of the 2008-2009 school year. There were 30 million training rows and 1.2 million test rows in total. For this project, we are using the Algebra-2008-2009 dataset with information recorded for 575 students for 809,695 steps overall using the tutoring system. Overall, the dataset contains 18 features for prediction with 6 categorical, 7 sequential, 4 timestamps, and the final one, the label, binomial. We have used training data as the complete dataset for this project by splitting it into training and testing dataset since we did not have true predictions for the original challenge Algebra I 2008-2009 Dataset to find the testing accuracy.

	Data sets	Students	Steps
Development	Algebra I 2005-2006	575	813,661
	Algebra I 2006-2007	1,840	2,289,726
	Bridge to Algebra 2006-2007	1,146	3,656,871
Challenge	Algebra I 2008-2009	3,310	9,426,966
	Bridge to Algebra 2008-2009	6,043	20,768,884

Table 1: KDD Cup 2010 Data Sets (PSLCDataShop, 2010b)

The data shows records of student interactions with the tutoring system while solving various problems for each unit to master certain set of skills with description of important fields as follows:

- **Problem:** It is a task for a student to perform that typically involves multiple steps. There are many such tasks for every unit/topic covered by the tutor for the students learning. There were 20 unique problems solved by different students.

- **Step:** To solve the given problem, the student has to provide a step-by-step solution to reach the answer. Each step in this solution is recorded sequentially and matched against a predefined set of steps.

- **Transaction:** Each attempt, event, correct/incorrect action taken by the student.

- **Knowledge Component:** The skill-set/knowledge required to perform a certain step correctly to go forward towards the answer. If a student is able to perform a certain step, it means that the student possesses those knowledge components related to that step.

- **Opportunity:** An opportunity is a chance for a student to demonstrate whether he or she has learned a given knowledge component.

knowledgeComponent	stepStartTime	errorStepDuration (sec)
anonStudentId	firstTransactionTime	correctFirstAttempt
problemHierarchy	correctTransactionTime	incorrects
problemName	stepEndTime	hints
problemView	stepDuration (sec)	corrects
stepName	correctStepDuration (sec)	opportunity

Table 2. Fields in the provided dataset used for data preprocessing steps

3.2 Data Processing

For any data mining problem, data preprocessing is one of the very crucial steps since the quality of data that is being fed into the model can affect the model performance. Hence data cleaning, transformation and handling missing values becomes important to have a good and clean dataset to feed into our algorithm. The details about the training dataset are given in the section above, so we are performing following two steps on that data as part of data preprocessing phase which would then be fed into the next stage of Feature selection to extract relevant features from the dataset.

3.2.1 Data Cleaning

The data is cleaned to remove redundant information like the description of a skill in the Knowledge Component to just extract the corresponding skill value. For example, [Skill-Rule: Remove constant; ax+b=c, positive; ax+b=c, negative; x+a=b, positive; x+a=b, negative; [var expr]+[const expr]=[const expr], positive; [var expr]+[const expr]=[const expr], negative; [var expr]+[const expr]=[const expr], all; Combine constants to right; Combine constants to left; a-x=b, positive; a/x+b=c, positive; a/x+b=c, negative] is cleaned to have Remove Constant as the value in the field. Similarly, we have broken down the value of the field Problem Hierarchy to Unit name and problem number to have a proper record of the problem that the student is solving using the tutor. For example, Unit ES 04, Section ES 04-15 is broken down to ES04 as Unit and 15 as the problem number.

3.2.2 Handling Missing Values

Step start time is the field that has blank values for 919 records among the timestamped fields. To handle missing values for them, we are adapting following two approaches inspired from Zachary et. al [2]- if First Transaction Time is given, we are setting start time of the given step to be equal to that; otherwise we are taking the step start time to be equal to the end time of the previous step attempted by the student.
Around 25851 records have Correct Transaction Time as empty. We are not doing anything for this field since the missing value in this implies that student did not attempt this step correctly. So we are just setting it to 00:00.
1230 records have Step Duration as 0 which is either due to missing Step Start Time or when Step Start Time is equal to Step End Time. So after we get the amputated value for Step Start Time as discussed above, we calculate Step Duration.

3.2.3 Data Transformation

The categorical data like Skills obtained from KC field are converted to integer values by giving every skill/set of skills a unique identifier. Approximately 202,669 records do not have KC values, so we are treating that kind of skill as a new skill with another unique identifier. Also, there are rows with multiple KC and opportunity values. We are treating this set of skill (A,B) as a new skill set because if a step requires both the skills to be solved, then just assuming one of them to be present will not be sufficient.

After data processing and cleaning, we had the records of 573 students for 20 unique problems with total records of 689,505 transactions. 103 unique knowledge components were constructed after data transformation.

3.3 Feature Selection

Feature selection step can help us in identifying the set of features that have the biggest impact on our final output. The objective of this step is to resolve the data dimensionality problem by reducing the high dimensional data without losing its reliability for the next step of classification. We have done automatic feature extraction using Step Forward Selection algorithm which incrementally adds features to the model and then select the most significant set of features with the highest accuracy. Hence, instead of occurring as an independent process taking place prior to model building, this algorithm attempts to optimize feature selection for the given machine learning algorithm. We have used SequentialFeatureSelector from mlxtend library.

The final features extracted by the algorithm were - anonymous student ID, knowledge component, correct step duration with accuracy of 0.98%. Feature selection is done for each student model that is built in next stage.

3.4 Support Vector Machine modeling

This step shortlists the SVM kernel to be used for model building to predict the performance of the given student on a given step. SVM is known to be one of the good classification algorithms to work on linear and non-linear data by searching for a linear optimal separating hyperplane which is also called the decision boundary to separate one class from another using support vectors. We decided to use SVM on this problem because it has been successfully applied before [6], [12], [13] to predict student grades in academia but was not applied on the KDD Cup Dataset. We started by building three different models for SVM kernels - linear, rbf and sigmoid with default parameters. We modified the values of gamma and cost parameters. Gamma is the parameter used for non-linear hyperplanes. The higher the gamma value, the more it tries to fit the data to the model. We have set gamma as auto for our evaluations. Cost (C) is the penalty parameter of the error term. It controls the tradeoff between smooth decision boundary and classifying training points correctly. It is set to 1 for the evaluations after trying different values manually. The dataset is split for every student using 80-20% splitting to train and predict using each of these models. The results for the three models is shown in the figures below:

(In percentage)	Linear SVM Kernel	RBF SVM Kernel	Sigmoid SVM Kernel
Training Accuracy	79	93.7	78.9
Testing Accuracy	80	80.7	78.3

Table 3. Training and Testing accuracies for the three SVM kernels for comparison of models

We decided to use Linear SVM kernel since it gave us the best training and testing accuracy. RBF model seemed to overfit the data, hence we discarded its use, inspite of better numbers. The prediction accuracies for all the three models were close. The data for each student is picked and split into training and testing set, and the features are selected by step forward selection algorithm mentioned in the above section. For some models, both knowledge component and correctStepDuration is used for some only one of the feature giving the best accuracy is employed. Thus, linear SVM model is trained for each step that a student performs for the problems. The model then predicts if the student will perform the given math step correctly or not by taking into account his previous actions.

3.5 Bayesian Knowledge Tracing building

Bayesian Knowledge Tracing (BKT) is an algorithm used in many intelligent tutoring systems to model each users learning mastery of skills or knowledge components. It assumes that the student knowledge is represented as a set of binary variables, one per skil, where the skill is either mastered by the student or not. The observations in BKT are also binary - either a student gets a problem/step right or not. Given below is the description of the default parameters that are initialized before BKT is run and the probability equations to calculate the prediction for student learning the given skill:

- $p(L_0)$ or $p - init$, the probability of the student knowing the skill beforehand.
- $p(T)$ or $p - transit$, the probability of the student demonstrating knowledge of the skill after an opportunity to apply it
- $p(S)$ or $p - slip$, the probability the student makes a mistake when applying a known skill
- $p(G)$ or $p - guess$, the probability that the student correctly applies an unknown skill (has a lucky guess)

Equation (a):

$$p(L_1)_u^k = p(L_0)^k$$

Equation (b):

$$p(L_{t+1}|obs = correct)_u^k = \frac{p(L_t)_u^k \cdot (1 - p(S)^k)}{p(L_t)_u^k \cdot (1 - p(S)^k) + (1 - p(L_t)_u^k) \cdot p(G)^k}$$

Equation (c):

$$p(L_{t+1}|obs = wrong)_u^k = \frac{p(L_t)_u^k \cdot p(S)^k}{p(L_t)_u^k \cdot p(S)^k + (1 - p(L_t)_u^k) \cdot (1 - p(G)^k)}$$

Equation (d):

$$p(L_{t+1})_u^k = p(L_{t+1}|obs)_u^k + (1 - p(L_{t+1}|obs)_u^k) \cdot p(T)^k$$

Equation (e):

$$p(C_{t+1})_u^k = p(L_t)_u^k \cdot (1 - p(S)^k) + (1 - p(L_t)_u^k) \cdot p(G)^k$$

Assuming that the default parameters are set for all skills, the above formulas are used as follows: The initial probability of a student u mastering skill k is initialized by equation (a). If the student u learns and applies the skill k correctly, the conditional probability is calculated using equation (b) and in case he applies the skills incorrectly, equation (c) is used. This conditional probability is used to update the probability of skill mastery calculated by equation (d). Then the probability of the student correctly applying the skill on a future practice is calculated with equation (e).

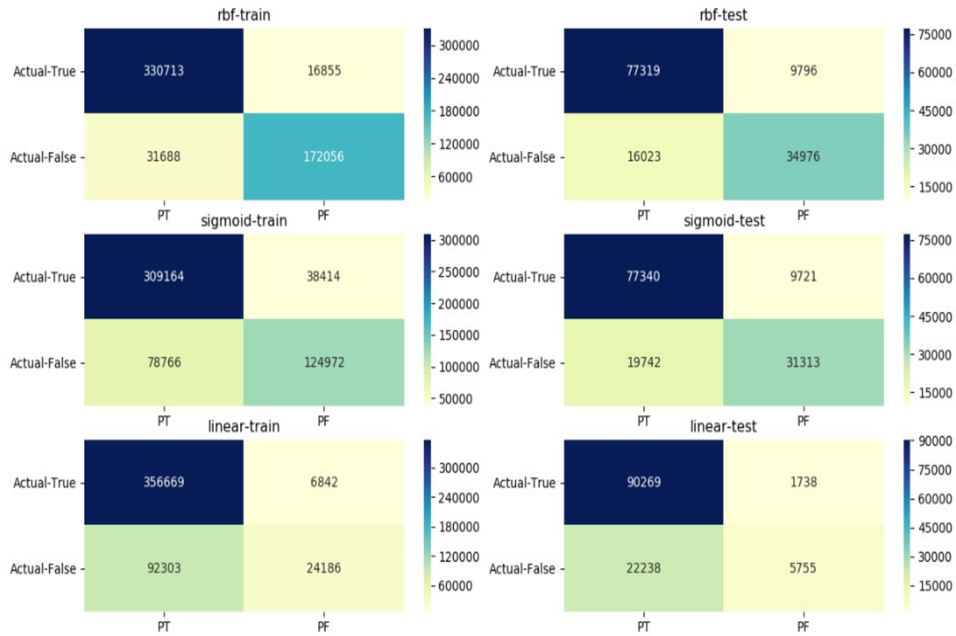
We developed the BKT model using the probability equations given above and tested for accuracy of the same with following default parameters by manual analysis of achieveing best accuracy: Initial: 0.78, Transit: 0.87, Guess: 0.32, Slip: 0.31.

4. Results and Analysis

The following results were obtained for Linear SVM Kernel:

	Accuracy	**RMSE**
Training	79.3%	0.19
Testing	80%	0.18

The confusion matrix for Linear SVM is:



We achieved the accuracy of 78.98% for Bayesian Knowledge Tracing model setting threshold as 0.5.

As noted in the observations above, linear SVM kernel performs slightly better than BKT model with accuracy difference of 1.02%. Since most of the research using KDD Cup 2010 Dataset did not make use of Support Vector Machines, inspite of them being one of the popular machine learning algorithm for predicting student performance, we wanted to see how this dataset performs for SVM independently. Now, that we have the numbers, we can say that SVM performs good but it can be further improved by combining it with other learning techniques to achieve a higher gain in accuracy. As far as comparison for BKT and SVM goes. We decided to compare the performance of SVM with BKT because BKT is the underlying prediction algorithm which is being used in Cognitive Tutors (the ITS that has been used to collect this data). We understand that both the algorithms are different in nature with BKT only making use of knowledge components that a student learns with time while SVM works like a typical machine learning algorithm with choice between knowledge component and correct step duration as its primary features. The RMSE values for KDD Datacup challenge winners was 0.2 but our RMSE value for linear SVM kernel cannot be directly compared with them because the challenge dataset was different from the one that we have used for prediction in our project.

5. Conclusion

The use of Intelligent Tutoring Systems is becoming highly prominent in the educational field. Such an extensive use of ITS gives us a chance to collect massive amount of student data and understand their learning processes. Hence, we can apply different learning techniques to predict the student performance. The strong performances of Bayesian Knowledge Tracing and Support Vector Machines in theory and on system help the Intelligent Tutoring System in creating the knowledge models of the students and provide insight into predicting the future outcomes of them solving progressive questions depending on the trained knowledge components. Although both the techniques have different approaches of training and implementation - BKT learns sequentially without data split for each student model, and SVM model learns from the subset of student data. So, comparing them directly wouldnt be fair but we can comment on their effectiveness with both of them providing greater insights into the students knowledge state.

Github Link: https://github.com/doneria-anjali/student-performance-predictor

References

[1] Yu, Hsiang-Fu et al. Feature Engineering and Classifier Ensemble for KDD Cup 2010. (2010)

[2] Pardos, Zachary A. and Neil T. Heffernan. Using HMMs and bagged decision trees to leverage rich features of user and skill from an intelligent tutoring system dataset (2010)

[3] W. Ham al ainen, M. Vinni, Comparison of machine learning methods for intelligent tutoring systems, in: Intelligent Tutoring Systems, Springer, 2006, pp. 525534

[4] S. Sembiring, M. Zarlis, D. Hartama, S. Ramliana, E. Wani, Prediction of student academic performance by an application of data mining techniques, in: International Conference on Management and Artificial Intelligence IPEDR, Vol. 6, 2011, pp. 110114

[5] G. Gray, C. McGuinness, P. Owende, An application of classification models to predict learner progression in tertiary education, in: Advance Computing Conference (IACC), 2014 IEEE International, IEEE, 2014, pp. 549554

[6] Eashwar, K. B., R. Venkatesan, and D. Ganesh. "Student Performance Prediction Using Svm." International Journal of Mechanical Engineering and Technology 8.11 (2017): 649-662

[7] C. Marquez-Vera, C. Romero, and S. Ventura, Predicting school failure using data mining, in Proc. 2011 4th Int. Conf. Edu. Data Mining, 2010.

[8] Y.-H. Wang and H.-C. Liao, Data mining for adaptive learning in a TESLbased e-learning system, Exp. Syst. Appl., vol. 38, no. 6, pp. 64806485, 2011.

[9] N. Thai-Nghe et al., Multi-relational factorization models for predicting student performance, in Proc. KDD Workshop Knowl. Discovery Edu. Data, 2011, pp. 2740.

[10] A. Toscher and M. Jahrer, Collaborative filtering applied to educational data mining, in Proc. KDD cup workshop: Knowledge discovery in educational data, 2010, pp. 1323.

[11] C. Tekin, J. Yoon, and M. van der Schaar, Adaptive ensemble learning with confidence bounds, in IEEE Trans. Signal Process, vol. 65, no. 4, pp. 888903, Feb. 2017.

[12] Al-Shehri, H., Al-Qarni, A., Al-Saati, L., Batoaq, A., Badukhen, H., Alrashed, S., ... Olatunji, S. O. (2017, April). Student performance prediction using Support Vector Machine and K-Nearest Neighbor. In Electrical and Computer Engineering (CCECE), 2017 IEEE 30th Canadian Conference on (pp. 1-4). IEEE.

[13] Oloruntoba, S. A., and J. L. Akinode. "INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES RESEARCH TECHNOLOGY STUDENT ACADEMIC PERFORMANCE PREDICTION USING SUPPORT VECTOR MACHINE."