Class vs. Student in a Bayesian Network Student Model

Yutao Wang and Joseph Beck

Worcester Polytechnic Institute
{yutaowang,josephbeck}@wpi.edu

Abstract. For decades, intelligent tutoring systems researchers have been developing various methods of student modeling. Most of the models, including two of the most popular approaches: Knowledge Tracing model and Performance Factor Analysis, all have similar assumption: the information needed to model the student is the student's performance. However, there are other sources of information that are not utilized, such as the performance on other students in same class. This paper extends the Student-Skill extension of Knowledge Tracing, to take into account the class information, and learns four parameters: prior knowledge, learn, guess and slip for each class of students enrolled in the system. The paper then compares the accuracy using the four parameters for each class versus the four parameters for each student to find out which parameter set works better in predicting student performance. The result shows that modeling at coarser grain sizes can actually result in higher predictive accuracy, and data about classmates' performance is results in a higher predictive accuracy on unseen test data.

Keywords: Bayesian Networks, Knowledge Tracing, Individualization, student-skill model, class-skill model.

1 Introduction

Student modeling is crucial for Intelligent Tutoring Systems (ITS) to improve and to provide better tutoring for students. For decades, researchers in ITS have been developing various methods of modeling students. Two of the most popular approaches are Bayesian Knowledge Tracing (KT) [1], which uses a dynamic Bayesian Network to model student learning, and Performance Factor Analysis (PFA) [2], which uses a logistic regression to predict student performance. Both techniques have a similar underlying assumption that two things are needed to model the student: one component concerns the domain, such as skill information in KT and PFA models, or item information in the PFA model; the other component is the student's problem solving performance on the skill.

However, there are other sources of knowledge that are not utilized, such as the performance of other students in the same class. Instead, only *this* student's previous performances are taken into account. Imagine there is a class of 20 students, 19 of whom get the first item on a skill wrong, and you want to predict the performance of the 20th student's first item on the skill. Intuitively, predicting that this student would also respond incorrectly seems like a safe bet. However, current student models such

K. Yacef et al. (Eds.): AIED 2013, LNAI 7926, pp. 151–160, 2013.

as KT and PFA will not be affected by those 19 incorrect responses, as they were all made by other students. What would the effect on predictive accuracy be if which class a student is currently in was factored into student models? Our intuition is that class perhaps contains important information such as the student's prior knowledge about a skill. Since all students in a class share a common teacher, curriculum, and assigned homework problems, we should expect similarities in performance. Our goal is to capitalize on this dependency to improve student modeling.

In fact, the US Institute for Educational Sciences requires grant proposals' power analyses to discount the sample size if there are multiple students in the same classroom, due to their lack of independence from each other (most statistical tests require each sample to be independent). Given that we know this dependence effect exists statistically, why not make use of it? In this paper, we are focusing on utilizing the class information to improve student modeling and trying to determine under which circumstances, using other students' information could be more beneficial than using current student's individual information.

Section 2 introduces the model and dataset we are using in our experiments. Section 3 shows the experimental results. In section 4 and 5 we discuss the conclusions and future directions for our work.

2 Approach

This section briefly introduces the Student Skill model and the modification of it in order to allow class level individualization. The modified model also allows us to run experiments on various combinations of student and class information to determine whether or not the class information is better than the student information for each parameter.

2.1 Model

Knowledge Tracing is one of the most popular methods for modeling student knowledge. The original Knowledge Tracing model do not allow for individualization, and assumes that all students have the same probability of knowing a particular skill at their first opportunity, or slipping (making a careless mistake) on a skill, or learning a particular skill. This assumption is almost certainly invalid, as students are likely to differ in these aspects. Several researchers have tried to show the power of individualization [4, 5]. The model we use in this work is build upon one of the individualization model called the Student Skill model [4]. The idea of the Student Skill model is that rather than estimating a learning rate for each skill, instead view learning rate as being a function of the skill and of this individual learner. Perhaps some skills are learned more quickly or slowly than others, and perhaps some students learn more quickly or slowly than others. By combining both effects, it is possible to more accurately model the student.

The Student Skill model structure is shown in Fig.1. The goal of the Student Skill model is to add individualization into the original Knowledge Tracing model. It can learn four student parameters and four skill parameters simultaneously. The lowest two levels of this model is the same as the original Knowledge Tracing model (nodes

K1…Kn and Q1…Qn in Fig. 1). The Student Skill model adds upper levels to represent the student and skill information and their interaction. Two multinomial nodes are used to represent the identity of each student (node St in Fig.1) and each skill (node Sk in Fig.1). Instead of pointing the student identity and the skill identity nodes directly to the knowledge nodes, which will result in an exponentially increasing number of parameters, we instead added a level of nodes to represent the four student parameters (node StP, StG, StS and StL in Fig.1) and the four skill parameters (node SkP, SkG, SkS and SkL in Fig.1). Those parameter nodes are binary nodes which represents the high/low level of the corresponding parameters. For example, if the StP node is 1 for a student, means the student has high level of prior knowledge, and if the StP node is 0 for a student, means the student has low level of prior knowledge. Then the next level combines the influence of the student parameters and the skill parameters and generated four standard Knowledge Tracing parameters (node P, G, S and L in Fig.1) to be used in the lowest two levels. In this way, we generate a knowledge tracing model that is custom-fit to each learner and for each skill.

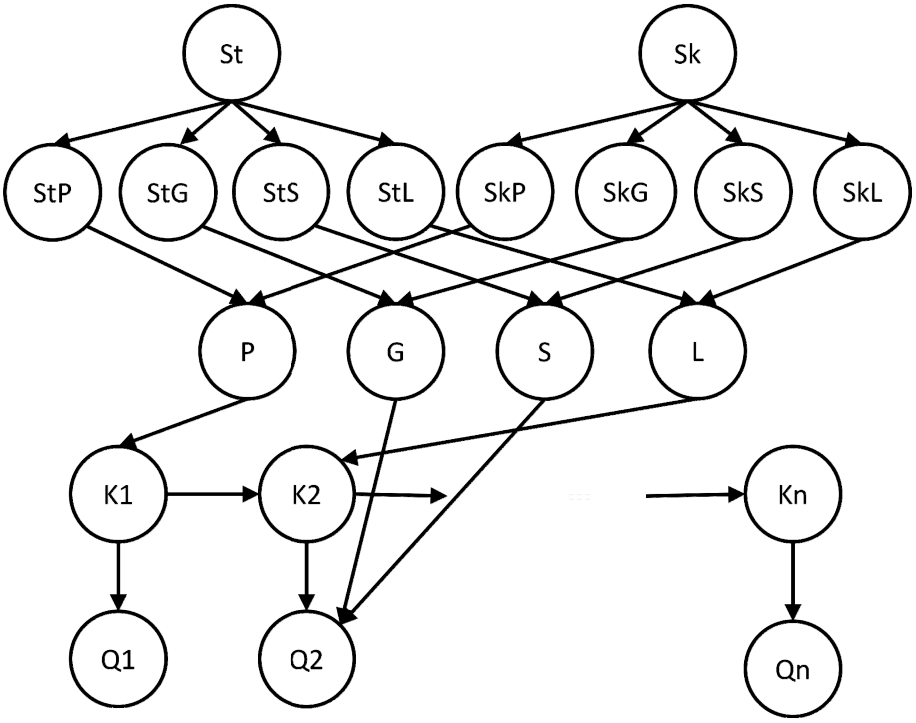

Fig. 1. The Student Skill model

One drawback of the Student Skill model is that it requires a large number of parameters. In addition to estimating four parameters per skill, it must also estimate four parameters per student. Given that many datasets have considerably more users than skills, this inflation in the number of parameters is a large concern. Therefore, we considered methods for reducing the number of parameters in our model, to enable them to better

generalize to unseen data. One approach is, rather than modeling the students as individuals, to instead model which mathematics class the student is enrolled in. Students within the same class have the same teacher, textbook, homework, and may even be grouped by ability in the subject. Given that, in our datasets, there are typically about 24 students per class, modeling class-level effects has 24 times as much data to estimate parameters. In addition, if we only model class parameters, we only have to estimate 1 set of parameters for each *class* of students, rather than 1 set for each individual students. Thus, the use of class information can be seen as a coarser grain-size individualization compared to the Student Skill model. We demonstrate the Class Skill model in figure 2, and the nodes are identified as follows:

- St: A multinomial node represents each student's identity, observable.
- Sk: A multinomial node represents each skill's identity, observable.
- StP: Student Prior Knowledge, binary node, latent.
- StG: Student Guess rate, binary node, latent.
- StS: Student Slip rate, binary node, latent.
- StL: Student Learning rate, binary node, latent.
- SkP: Skill Prior Knowledge, binary node, latent.
- SkG: Skill Guess rate, binary node, latent.
- SkS: Skill Slip rate, binary node, latent.
- SkL: Skill Learning rate, binary node, latent.
- P: Prior Knowledge of a particular student and a particular skill, binary node, latent.
- G: Guess rate of a particular student and a particular skill, binary node, latent.
- S: Slip rate of a particular student and a particular skill, binary node, latent.
- L: Learning of a particular student and a particular skill, binary node, latent.
- K1~Kn: Knowledge, binary node, latent.
- Q1~Qn: Question performance, binary node, latent.

The Student Skill model can easily be changed to consider the class information rather than the student information by replacing the St node to be a class node (Cl), and the parameters StP, StG, StS and StL will be turned into class prior (ClP), class guess (ClG), class slip (ClS) and class learning rate (ClL).

Instead of simply using class information to replace the student information, which is still considering only one resource of information, this paper combines these two models together to explore whether knowing which class a student is in is a better predictor than knowing which student, for each parameter in the model. For example, perhaps slip rate is best modeled at the individual student level, while learning rate is best estimated at the class level? Therefore, we have run experiments with different ways of combine the two resources of information trying to determine which parameter is best modeled using which source of information.

As shown in Fig. 2, the model is almost the same as the Student Skill model in Fig. 1. The only difference is the addition of the class (Cl) node, which is a multinomial node, represents which class a student is in. Nodes StP, StG, StS, StL turns into StP/ClP, StG/ClG, StS/ClS, StL/ClL, which means the nodes can either be a student level parameter or a class level parameter. The dash line between node Cl and node StP/ClP is a potential relationship in the model, as well as the dash line between node St and node Stp/ClP. If we choose one of these two dash lines, the other one will be ignored as if it does not exist. For example, if we choose to use class information for

prior knowledge, the dash line between St and node StP/ClP is ignored, and the node StP/ClP only contains the class prior (ClP). The same assumption is hold for all the other dash lines and parameters of class and student: StS/ClS, StG/ClG, StL/ClL.

Based on this model, by choosing different dash lines, we can test the best combination of class and student parameters and find the variability.

In our experiment, we used the Bayes Net Toolbox for Matlab developed by Murphy [6] to implement the Bayesian network student models and the Expectation Maximization (EM) algorithm to fit the model parameters to the dataset. The EM algorithm finds a set of parameters that maximize the likelihood of the data by iteratively running an expectation step to calculate expected likelihood given student performance data and a maximization step to compute the parameters that maximize that expected likelihood.

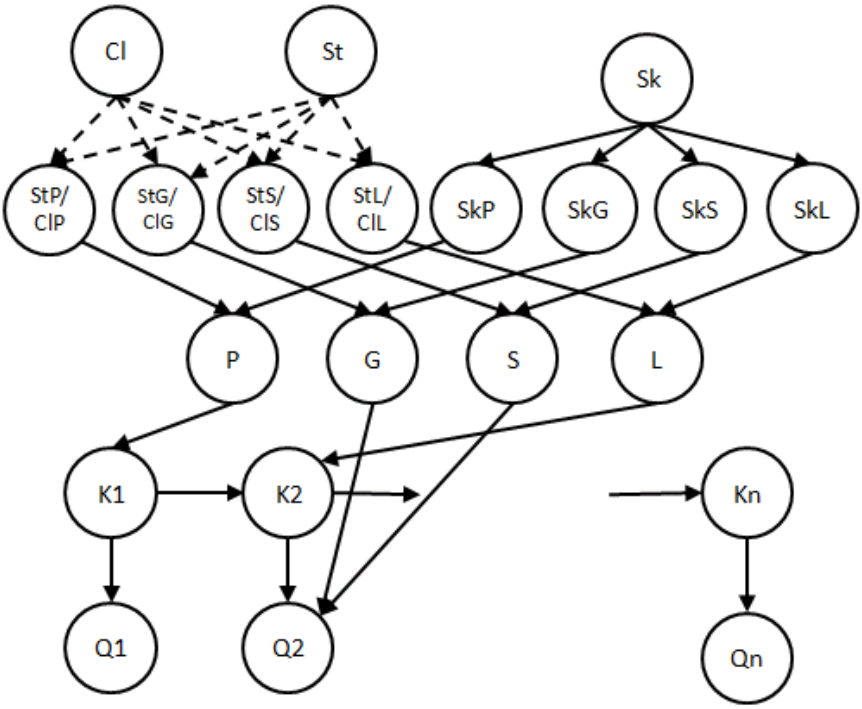

Fig. 2. Combination of Class Skill model and Student Skill model

2.2 Data and Model-Fitting

The data used in the analysis presented here came from the ASSISTments platform (www.assistments.org), a freely available web-based tutoring system for 4th through 10th grade mathematics. The performance of a question is marked as wrong if the first response is incorrect, or if the student asks for help.

We randomly sampled data of one hundred 12-14 year old 8th grade students from 4 classes and fifty skills from the school year September 2010 to September 2011. There are in total 53,450 problem solved in the dataset.

To make sure there were sufficient data in the training set to estimate parameters for students and skills, we divide the dataset into a training set and a test set using the following strategy: for each student, for every skill that she was practicing we flipped a coin and assigned this student-skill pair into either the training set or into the testing set. This process enables us to have a broad coverage of students and skills in the training set, to enable generalization to the testing set. However, we do not have data for the same student-skill pair in both the training and in the testing data. In this way, we maintain a relatively independent test set, but still enable our approach to see enough types of data to estimate all of the required parameters.

In the experiment, we estimate each knowledge tracing parameter using data about the skill, and either data about this student's or the student's classmates' performance on this skill. Thus, for each parameter we tried two ways of estimating its value. We examined each combination of settings for all four knowledge tracing parameters (P,G,S,L) To simplify the problem, we group the performance parameters, guess and slip, together. This leaves us in total 2^3= 8 different combinations in parameters. The models and experimental results are shown in the next section.

3 Results

The accuracy of the predictions was evaluated in terms of the Root Mean Squared Error (RMSE), with lower values meaning higher accuracy. We compared different models to analyze the best individualization level for prior Knowledge (K0), learning rate (L) and Guess and Slip (G/S) respectively. That is, for each of the parameters (K0, L, G/S), we choose Class level individualization or Student level individualization, there are in total 8 possible combinations. The different combination models and their RMSE results on the test set are shown in Table 1.

The first column shows which parameter is chosen for the prior knowledge, the second column shows which parameter is chosen for the learning rate, the third column shows which parameter is chosen for the performance parameters (guess and slip), the fourth column shows the RMSE result of each model on the test dataset. We order the rows in this table based on the RMSE on the test set, with the top rows representing higher accuracy on the test set.

Table 1. RMSE result on test and training data

K0	L	G/S	RMSE
Class	Student	Class	0.413
Class	Class	Class	0.415
Class	Student	Student	0.417
Class	Class	Student	0.419
Student	Student	Student	0.421
Student	Student	Class	0.423
Student	Class	Class	0.424
Student	Class	Student	0.425

For comparison, the standard Knowledge Tracing model produces an RMSE of 0.428 on the test data, which is less accurate than all of the models we experimented with in Table 1. Therefore, it appears that both of the class level and the student level individualization can help improve Knowledge Tracing's predictive accuracy.

A second point of comparison is our baseline Student Skill model, represented in the 5th row in this table (underlined), which represents estimating all of the parameters using information about each student. Thus, each student has a customized estimate of prior knowledge (K0), learning (L), and guess (G) and slip (S), as they are derived from the student node. In this case, model in Fig. 2 degenerates to be the same as the Student Skill model in Fig. 1. The fact that this model is only at the middle of the table shows that, it is not as strong as other methods of estimating parameters.

In other words, sometimes it is better to use the class information rather than using individual student information. This result could occur if students within a class do not vary very much on a particular parameters. In that case, it would be better to estimate that parameter for the entire class to take advantage of the larger quantity of data. For example, the fact that the 4th row, which has prior and learning comes from class information, and guess and slip comes from the student information results in lower RMSE value on the test data than the 5th row, indicates that the prior knowledge and learning rate may be better estimated through the class information rather than estimated from completely individualization of student. Back to the example at the beginning of this paper, this means that for prior knowledge, and guess and slip rate, knowing the information of all of the other students in the class may be slightly more beneficial than only knowing the information of the current student. If all of the other students in the class do not know a skill initially, it is more likely the current student do not know the skill either, no matter how good the student is on other skills.

Among all of these models, the best mode (the first row in the table) is the one with prior knowledge (K0) and performance parameters (guess and slip) derived from the class information, and the learning rate (L) is derived from individual student information. The result seems plausible because all students in a class normally get the same instruction, thus might have similar prior knowledge (K0) about a particular skill, and some students learn faster than others, thus the learning rate (L) would be beneficial from individual student information. To be clear, we are not asserting that all students have the same prior knowledge, as some students will not complete homework or might not pay attention in class. However, within a class, prior knowledge varies less than the other parameters, and, at least in this instance, the potential benefit of customizing K0 to each student is not worth the additional parameters.

Besides finding the best combination of grain-sizes for estimating various parameters, there are also some interesting general trends visible in Table 1. The most interesting one is that prior knowledge (K0) is always better modeled at the class level: the top 4 rows are all with class information used to estimate the K0 parameter. This result confirms our intuition that all students in a class tend to have similar prior knowledge, which could be caused by the fact that they are going through similar instructions, or the fact that similar students are tend to be assigned to the same classroom.

The trend in learning rate (L) is the opposite as the trend for prior knowledge. Since the bottom two rows both have class information as the resource for learning rate, student information seems to be a more powerful resource. Therefore, within a class, students' ability to learn mathematics appears to vary more than their prior

knowledge. However, these differences appear to be rather small: comparing the first and second lines results in a difference in RMSE of 0.002; similarly, comparing the third and fourth lines also results in a difference in RMSE of 0.002. This difference is rather small, so estimating learning rate at the class level or at the student level works approximately equally well.

As for the performance parameters (guess and slip), there seems to be a general advantage to modeling these effects at the class level, but the trend is not completely clear. We expected guess and slip behaviors to vary considerably within a class, and to be better modeled at the student level. Therefore, we found this result somewhat surprising.

4 Contributions, Future Work, and Conclusions

This paper makes three main contributions. Philosophically, it considers the learner's classmates as a viable source of information for predicting the learner's behavior. This source of information seems to have been overlooked by the ITS community.

The second contribution this paper makes computational, as it extends the Bayesian knowledge tracing framework to take into account the class information. Our model structure enables us to model parameters at the class- or student-level, and to mix and match grain sizes within an experiment. In a similar effort, a PFA-like model was modified to account for class-level information [7].

The third contribution this paper makes is empirical. Our results suggest that initial knowledge of a skill is probably best modeled at the class level. Prior work either assumed the initial knowledge is determined either by the skill itself or a combination of the student and skill. This paper's experimental results suggest that student modelers should consider additional sources of power for understanding learners.

Currently, the way we utilize the student and class information is to consider using either class parameters or student parameters. That is, each of the models we compared considered using one source of power for each of the parameters, but not both. It is possible that we can look at both sources information simultaneously and even take into account the fact that a student is a member of a class, to build a hierarchically structured model that blends the two sources of information together. In this model, class could be the parent node of different students. The model is easy for people to understand and interpret, yet we are not sure if a complex Bayesian Network representation of this model can be properly built and learn back the expected parameters. Both experiments with real and simulated data will be helpful for evaluating such approaches. It is also unclear if the model will be practical given the large number of parameters required.

One issue that we have not yet addressed is whether the performance parameters (guess and slip) should be grouped together. In this paper, we group the performance parameters together to simplify the experiments based on the assumption that these two parameters are both related to performance and should have similar properties with respect to the best grain size for modeling. Yet, it is likely that guess and slip behaves very differently at the class level compared to the student level. For example, some type of instruction may cause all students in the class very likely to guess the correct answer for some skills, even though the students do not fully understand the

skill. We suspect that slip is best modeled at the individual level. The mixed result in the performance parameters in section 3 could perhaps become more clear if we run more experiments with separate guess and slip parameters.

Another question that we are interested in exploring is whether the results about class-level parameters transfer across years? Currently, our evaluation looks at only one year's data and generates the test and training set from that year. This approach has the normal cold start problem, that if it is the start of a new school year and we have no information about the class yet, what would be a reasonable information to use to build the student model? One possible solution that we are interested in is to use the class information of previous school years. If we can find a class that we have data from previous years that is similar to a current class, we might be able to use the information from that class to start building the model for the current class. How to define similarity of different classes, however, is a challenging question. We could look at the teacher or use the very first performance of each student in the class as an estimate of prior knowledge. We could also choose a set of similar previous classes and use the average of their parameters instead of choose only one from all. Or, we could use whichever prior class has the highest predictive accuracy for this student, as in [3].

Finally, from a broader perspective, class can be seen as a group of students, thus is a natural way of clustering students. There are literatures that focus on clustering in student modeling such as [8,9]. What are the differences and connections between using class and using other clustering methods? Class could be an effect of the teacher or ability grouping; in this case, using clustering algorithms on features such as teacher and student ability could result in similar clusters as classes. There are also other levels of abstraction and natural clustering, such as which grade or school a student is in, exploring models that utilizing these new sources of information is also new and interesting.

In summary, this paper introduces a framework for using a dynamic Bayesian network to model parameters as a combination of student-skill effects, or class-skill effects. We have found that using either source of knowledge is more accurate than a standard knowledge tracing model. By selectively estimating some parameters at a coarser grain size, we are able to improve accuracy a bit over the class-skill model.

Acknowledgements. This work was supported by the National Science Foundation (grant DRL-1109483) to Worcester Polytechnic Institute. The opinions expressed are those of the authors and do not necessarily represent the views of the Foundation.

References

1. Corbett, A., Anderson, J.: Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. User Modeling and User-Adapted Interaction 4, 253–278 (1995)
2. Pavlik, P.I., Cen, H., Koedinger, K.: Performance Factors Analysis – A New Alternative to Knowledge. In: Proceedings of the 14th International Conference on Artificial Intelligence in Education, pp. 531–538 (2009)
3. Gong, Y., Beck, J.E., Ruiz, C.: Modeling Multiple Distributions of Student Performances to Improve Predictive Accuracy. In: Masthoff, J., Mobasher, B., Desmarais, M.C., Nkambou, R. (eds.) UMAP 2012. LNCS, vol. 7379, pp. 102–113. Springer, Heidelberg (2012)

4. Wang, Y., Heffernan, N.T.: The Student Skill Model. In: Cerri, S.A., Clancey, W.J., Papadourakis, G., Panourgia, K. (eds.) ITS 2012. LNCS, vol. 7315, pp. 399–404. Springer, Heidelberg (2012)
5. Pardos, Z.A., Heffernan, N.T.: Modeling Individualization in a Bayesian Networks Implementation of Knowledge Tracing. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 255–266. Springer, Heidelberg (2010)
6. Murphy, K.P.: The Bayes Net Toolbox for Matlab, Computing Science and Statistics. Proceedings of Interface 33 (2001)
7. Xiong, X., Beck, J.E., Li, S.: Class distinctions: Leveraging class-level features to predict student retention performance. In: Chad Lane, H., Yacef, K., Mostow, J., Pavlik, P. (eds.) AIED 2013. LNCS (LNAI), vol. 7926, pp. 820–823. Springer, Heidelberg (2013)
8. Trivedi, S., Pardos, Z.A., Heffernan, N.T.: The Utility of Clustering in Prediction Tasks. In: Proceedings of the 17th Conference on Knowledge Discovery and Data Mining (2011)
9. Song, F., Sarkozy, G.N., Trivedi, S., Wang, Y., Heffernan, N.T.: Applying Clustering to the Problem of Predicting Retention within an ITS: Comparing Regularity Clustering with Traditional Methods. Accepted by the 24th FLAIRS