Predictive Student Modeling in Educational Games with Multi-Task Learning

Michael Geden,[1] Andrew Emerson,[1] Jonathan Rowe,[1] Roger Azevedo,[2] James Lester[1]

[1]North Carolina State University, [2]University of Central Florida
{mageden,ajemerso,jprowe,lester}@ncsu.edu, roger.azevedo@ucf.edu

Abstract

Modeling student knowledge is critical in adaptive learning environments. Predictive student modeling enables formative assessment of student knowledge and skills, and it drives personalized support to create learning experiences that are both effective and engaging. Traditional approaches to predictive student modeling utilize features extracted from students' interaction trace data to predict student test performance, aggregating student test performance as a single output label. We reformulate predictive student modeling as a multi-task learning problem, modeling questions from student test data as distinct "tasks." We demonstrate the effectiveness of this approach by utilizing student data from a series of laboratory-based and classroom-based studies conducted with a game-based learning environment for microbiology education, CRYSTAL ISLAND. Using sequential representations of student gameplay, results show that multi-task stacked LSTMs with residual connections significantly outperform baseline models that do not use the multi-task formulation. Additionally, the accuracy of predictive student models is improved as the number of tasks increases. These findings have significant implications for the design and development of predictive student models in adaptive learning environments.

Introduction

Recent years have seen growing interest in modeling student knowledge in adaptive learning environments (Piech et al., 2015; Mao, Lin, & Chi, 2018; Gardner, Brooks, & Baker, 2019). Predictive student modeling is the task of predicting students' future performance on a problem or test based upon their past interactions with a learning environment. Predictive modeling is important for tailoring student experiences in a range of adaptive learning environments, such as intelligent tutoring systems (Gardner, Brooks, & Baker, 2019) and educational games (Shute et al. 2016; Min et al., 2019). By modeling student knowledge, adaptive learning environments can personalize delivery of problem scenarios, hints, scaffolding, and feedback to create student learning experiences that are more effective than one-size-fits-all approaches (VanLehn, 2011). However, predictive student modeling is a challenging machine learning task because student data is often noisy, heterogeneous, and expensive to collect (Bosch et al. 2016).

Predictive student models typically represent student knowledge as an aggregate across a student's performance on a set of questions. For example, a typical output label in predictive student modeling is the overall accuracy of student responses on a *post-test* administered after the student has finished interacting with an adaptive learning environment. This approach makes stringent assumptions that each post-test question has an equivalent mapping from features in the input space and is equally representative of the underlying latent construct being measured (e.g., science content knowledge). A natural extension is to relax these assumptions by employing multi-task learning (MTL), wherein each test question is an outcome variable in the same predictive model. MTL has been shown to yield improved model accuracy across a range of domains by sharing feature representations across different tasks, which provides a natural form of model regularization (Zhang & Yang 2017; Argyriou, Evgeniou, & Pontil 2007). MTL has particular promise for predictive student modeling, where there are typically multiple test questions designed to assess the same knowledge and where there is often limited data available on student interactions with the particular adaptive learning environment.

In this paper, we present a novel predictive student modeling framework using MTL. We utilize MTL to model student outcomes at the item level within a game-based learning environment for middle school science education, CRYSTAL ISLAND. Empirical results

demonstrate the efficacy of the approach with markedly improved results over what is typical for predictive student modeling. Additionally, we explore how different mechanisms of self-attention can influence model performance through selecting relevant sections of student gameplay interactions.

Related Work

Student Modeling

A widely used approach for modeling student knowledge in adaptive learning technologies is Bayesian knowledge tracing (BKT) (Mao et al. 2018). BKT models student knowledge as a binary latent variable in a hidden Markov model. The model is updated based upon student interactions with an adaptive learning environment, which provide evidence of student knowledge and skills over time. Although BKT is an effective approach to student modeling in adaptive learning environments, it is not always well suited for student modeling in educational games, particularly in cases in which a game-based learning environment lacks repeated content exercises that provide recurring evidence of student skills.

An alternative to Bayesian knowledge tracing is *stealth assessment*, which utilizes methods from evidence-centered design to devise Bayesian networks that link student actions with content knowledge based upon network structures that are manually authored by domain experts (Shute et al. 2016). Stealth assessment is an effective approach for predictive student modeling in educational games, but the models are labor-intensive to construct. A related framework is *deep stealth assessment*, which utilizes long short-term memory (LSTM) networks to predict student test performance following interaction with an educational game and has shown promising results at modeling student knowledge without requiring domain experts (Min et al. 2019).

Item response theory (IRT) models the probability that a student will correctly answer a given exercise by incorporating the characteristics of both the test-taker and the questions (Embretson & Reise 2013). IRT does not assume all questions are the same difficulty, and it can model an individual's probability of success as a function of both their capability and the difficulty of the question. Extensions of this work include time-varying models (Lan, Studer, & Baraniuk 2014) and the integration of ideas from IRT into traditional BKT models (Khajah et al. 2014). More recent work has investigated recurrent neural networks to capture more complex representations of student knowledge and to estimate the probability that a student will answer the next question correctly (Piech et al. 2015). Other recent applications include the use of LSTM-based architectures with an attention mechanism to predict student performance for the personalization and sequencing of exercises (Su et al. 2018). Our work utilizes similar sequential architectures, but we incorporate methods from multi-task learning to significantly improve model performance.

Multi-Task Learning

Recent years have seen a growing interest in multi-task learning in applications such as computer vision (Fang, Zhang, Zhang, & Bai 2017; Kendall, Gal, & Cipolla 2018), climate modeling (Goncalves, Von Zuben, & Banerjee 2016), healthcare (Jin, Yang, Xiao, Zhang, Wei, & Wang 2017), and dialogue analysis (Tong, Fu, Shang, Zhao, & Yan 2018). Multi-task learning has been shown to improve model fitting by sharing information across multiple outcome variables, providing shared components of the model with additional training data and enhanced regularization (Zhang & Yang 2017). Multi-task learning dramatically reduces the number of parameters that need to be estimated, as well as the compute time required compared to running each outcome variable separately. This is operationalized by sharing weights across multiple tasks based upon the assumption that the tasks have an inherent relationship (Shui et al. 2019). A challenge of multi-task learning is the sensitivity to the choice of loss weights for each of the tasks. Hyperparameter tuning of the loss weights is effective with a small number of tasks but does not scale well as the number of tasks increases. An alternative approach is to estimate the loss weights as part of the model building process (Kendall et al. 2018).

Incorporated into adaptive learning environments in which data collection is often labor intensive compared to other machine learning applications. Therefore, frameworks that make efficient use of training data and incorporate regularization effectively can be beneficial in building predictive models from datasets with a limited sample size (Sawyer et al. 2018). Multi-task learning allows for the separate modeling of individual questions, which IRT has demonstrated can have largely different characteristics even if the questions are manifesting from the same underlying latent variable. A previous study found favorable results using MTL to predict student test scores using a standard feedforward neural network (Bakker and Heskes 2003), but it did not involve sequences of student actions as are often encountered in adaptive learning environments. Additionally, only one weighting of each tasks' loss function was explored, even though different loss weightings can have a large effect on model accuracy (Kendall, Gal, & Cipolla 2018).

Dataset

We investigated the multi-task learning framework for predictive student modeling in an educational game for microbiology education, CRYSTAL ISLAND (Rowe et al. 2011). In CRYSTAL ISLAND, students take the role of a medical field agent investigating an infectious outbreak on

a remote island (Figure 1). Students talk with non-player characters, explore different locations, read virtual books and microbiology posters, test hypotheses about the outbreak in a virtual laboratory, and record their findings in a virtual diagnosis worksheet. As students navigate through the game, their actions and locations are stored in trace log files that are subsequently used for modeling.

In this work, we used data from two different samples of students across different contexts (laboratory and classroom) to increase the heterogeneity of the sample and the generalizability of the resulting model (Sawyer et al. 2018). Students from both samples answered the same pre- and post-test surveys, but there were some differences in the experimental setup and game. Combining the data from the university-based laboratory study (n = 62) with the data from the classroom-based study (n = 119), the total sample size of the dataset is 181 students.

Prior to playing the game, students completed a pre-game survey containing demographic questions, questionnaires about student interest and achievement goals, and a 17-item microbiology content knowledge pre-test composed of multiple-choice questions. Each question had four options with one correct answer. The questions centered on microbiology content such as pathogens, viruses, carcinogens, and bacteria. Students then played CRYSTAL ISLAND until they either solved the in-game mystery or they ran out of time. After playing the game, students completed a post-game survey, which contained a separate set of 17 microbiology content knowledge questions. The post-test microbiology content items were summed to create a single post-test score.

Figure 1: CRYSTAL ISLAND Game Environment.

Feature Representation

The input features for all models consisted of items from two components of the pre-game survey (33 features), an indicator variable representing the dataset which the student belonged to (3 features), and the student's gameplay actions within CRYSTAL ISLAND (130 features), which yielded a total of 166 features. From the pre-game survey, we used 16-items from a survey on emotions, interest, and value (Likert scales) and a 17-item microbiology content pre-test (correct/incorrect answers).

Similar to previous work that used gameplay log features in a learning environment, we used a one-hot encoding of student actions using several components (Min et al. 2017):

- **Action type**: The system records each time the student moves to a new location within the virtual environment, engages in conversation with a non-player character (NPC), reads a virtual book or article, completes an in-game milestone (e.g., identifying the outbreak's transmission source), tests a hypothesis, or records findings in the diagnosis worksheet. The data include 8 distinct player action types.
- **Action arguments**: Action arguments are specific to the type of action the student is taking. For example, they include the name of the book the student is reading, the NPC with whom the student is conversing, and the object the student is testing in the virtual laboratory. The data contains 97 distinct types of player action arguments.
- **Location**: Within the game world, the system logs the location of each action. The data tracks 24 non-overlapping, discrete regions of the virtual game world.
- **Game time elapsed**: The system logs the time of each student action within the game, which is transformed into elapsed seconds since the start of gameplay.

Predictive Student Modeling with Multi-Task Stacked LSTMs

Student assessments are composed of multiple questions measuring the same construct (e.g., science content knowledge, personality) in order to provide accurate and reliable results. The traditional paradigm for modeling student assessments is to represent the outcome as an aggregate of the student's performance across all questions. This approach constrains the model to utilize the same feature encoding $f(x_i, \theta_1) \rightarrow h_i$ and mapping from the feature encoding $g(h_i, \theta_2) \rightarrow y_i$ across questions.

In this work, student knowledge modeling is reconceptualized within a multi-task learning framework, allowing for a shared feature representation for efficient estimation, but providing increased flexibility of different question characteristics through unique mappings from the encoding space. The long sequences of student actions generated from the game-based learning environment are modeled using a stacked LSTM with a residual connection. We explore how attention can potentially help the model focus on relevant sections of gameplay (Luong, Pham, and Manning 2015). Finally, the pretest data containing student characteristics is concatenated with the encoded gameplay features, fed into a dense layer, and then output as a prediction via the output layer.

Single-Task Learning

Consider a dataset composed of a d dimensional input space associated with a set of K correct/incorrect responses

to questions across n i.i.d. students. The performance of each student is represented as the sum of questions they answered correctly, $\tilde{y}$. If using mean-squared error as the loss function for $\tilde{y}$, this single-task representation has the following formulation:

$$L(\Theta) = \frac{1}{N}\sum_{i=1}^{N}\left(\tilde{y_i} - f(x_i, \theta)\right)^2$$

$$= \frac{1}{N}\sum_{i=1}^{N}\left(\sum_{k=1}^{K}\left(\widetilde{y_{i,k}} - \frac{f(x_i, \theta)}{K}\right)\right)^2$$

This formulation assumes that the loss for each of the T tasks are weighted equally. Additionally, each task is given an identical, shared representation, $f(x, \theta)$.

Multi-Task Learning

A multi-task learning framework relaxes the assumption that all tasks are weighted equally by having both a shared representation, $f(x, \theta)$, and a task specific representation, for each task K. The overall multi-task learning loss function is often composed as a weighted sum of the individual task loss functions:

$$L(\Theta) = \sum_{k=1}^{K} w_k \, L_k\left(y_k, g(f(x, \theta), \theta_k^*)\right)$$

The weight of each individual task, w_k, must be determined before training the MTL model and thus is not learned. A challenge stemming from this fact is that the overall loss can be sensitive to the selection of each w_k, which can become prohibitively expensive to tune as K grows large.

Uncertainty weighted. Kendall et al. (2018) proposed an alternative method for selecting w_k by estimating it as a parameter within the model. The form of the adjusted loss function is derived from the log-likelihood of the multivariate normal distribution based on an assumption of independence across tasks. In order to prevent the model from selecting $w_k = 0 : \forall k \in K$, an additional regularization term is added. Equal weighting across tasks is a special case of this formulation when $w_k = 1 : \forall k \in K$.

$$L(\Theta) = \sum_{k=1}^{K} w_k \, L_k\left(y_k, g(f(x, \theta), \theta_k^*)\right) - \log w_k$$

Self-Attention

Given a sequential output of length T of an m dimensional recurrent unit, $h_i \in R^{m \times T}$, the most common approach to obtaining a static representation is to either take the unweighted average across the entire sequence or to select the last output from the recurrent unit. An alternative approach is to use self-attention, where a weighted average is taken across the sequence. There are a number of approaches to estimate attention weights, a_i. Here we describe the multiplicative approach outlined in Luong, Pham, and Manning (2015), where $W \in R^{m \times m}$, $b \in R^m$, and $v \in R^m$ are estimated parameters.

$$l_i = v^T \tanh(W h_i + b)$$

$$a_i = \text{softmax}(l_i)$$

In addition to the traditional form of self-attention shown above, we also utilized a simplified form, given our smaller dataset, where $W \in R^m$ instead of an $m \times m$ matrix. This greatly reduces the number of parameters at the cost of limiting the flexibility of the model.

Implemented Predictive Student Model

To investigate MTL for predictive student modeling, we compared three model architectures: a single-task representation, an unweighted multi-task representation, and an uncertainty weighted multi-task representation. Each of the architectures were fit using three attention configurations: no attention, a simplified form of attention, and traditional matrix self-attention.

Single-task baseline. The single-task model utilized post-test score as the outcome variable with an identity activation function (Figure 2).

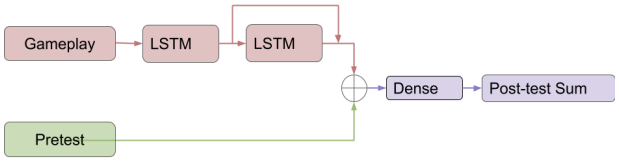

Figure 2: Single-Task Model Architecture.

Unweighted multi-task learning. The unweighted multi-task learning (MTL) model predicted the student's accuracy on each of the post-test questions, for a total of 17 tasks (Figure 3). Each question was modeled as a binary classification problem (i.e., correct/incorrect) with a sigmoid activation function. Binary cross-entropy was used as the loss function for each task. The relative weighting for each task's loss was selected prior to model training as a hyperparameter. Each task was weighted equally for the overall model's loss function.

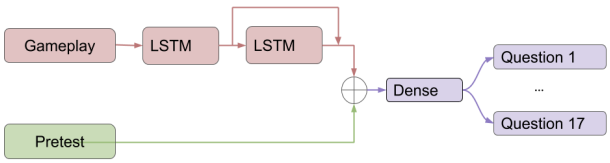

Figure 3: Multi-Task Model Architecture.

Uncertainty weighted multi-task learning. The uncertainty weighted MTL model predicted the student's

accuracy on each post-test question using a similar setup to the unweighted multi-task learning model (Figure 3). However, each task's relative loss weights were not preselected and were instead estimated as part of the model using the method outlined in Kendall et al. (2018).

Experiments

The single-task baseline models were formulated as a regression problem and trained to predict student post-test score. In contrast, the MTL models were trained as a joint binary classification problem across each of the 17 post-test items. The MTL predictions for each of the 17 items were summed to create a single post-test score in order to make comparisons with single-task baseline models. For the baseline models, we developed a set of predictive models utilizing a static representation calculated as the sum of each feature in the gameplay data in addition to a single-task neural network with an otherwise equivalent architecture to the MTL models. All models were trained and evaluated using 10-fold cross-validation along the same set of students to remove noise from sampling differences. In conducting the cross-validation, we ensured that no student data occurred both in the training and test sets. Hyperparameter tuning was conducted for each of the models within the 10-fold cross validation. Continuous data were standardized within each of the folds.

Static Models

A set of baseline models were selected using a static representation to assess if the added complexity of deep learning methods was beneficial over non-neural machine learning methods. The static baseline regression models for the single-task problem were the following: mean value, Lasso, Linear Kernel Support Vector Regression (SVR), Random Forest (RF), Gradient Boosting (GB), and multi-layer perceptron (MLP). In addition, we used a multi-task majority classifier baseline for each of the post-test items. Prior work on predictive student modeling in educational games has often utilized feature representations that consist of summary statistics describing students' gameplay behaviors (e.g., the number of books read, the number of laboratory tests run, etc.), which do not capture sequential patterns in student behavior over time (Sawyer et al. 2018). Student gameplay data was aggregated by summing the one-hot encoded variables of each student action across their total gameplay and dividing by their overall gameplay duration, resulting in their relative action rate.

Sequential Models

All sequential models were composed of two stacked long short-term memory (LSTM) layers with residual connections, a layer concatenating the LSTM gameplay features and pretest features, and a single densely connected layer (see Figures 2 and 3). The activation function for the dense layer, single-task output, and multi-task output were the hyperbolic tangent function, the identity function, and the sigmoid function, respectively. All models used early stopping using mean squared error with a patience of 15 and 500 maximum epochs. Every model was hyperparameter tuned using a grid search: number of LSTM units (32, 64, 128), number of dense units (32, 64, 128), and dropout rate (.33, .66). The best model was selected using the validation data and reported using the 10-fold test data.

Results

The lasso and random forest models tied for the best performance among the static baseline models (Table 1). The single-task models outperformed the static models by a moderate margin. The *no attention unweighted MTL* model and the *full self-attention weighted MTL* model tied for the best performance among the sequential models, with a large improvement over the single-task sequential baseline. Neither simple nor full attention had a notable effect on model performance with the exception of the weighted MTL model, where it provided a small improvement to model fit. All models terminated by early stopping prior to the maximum number of epochs.

The relationship between the number of tasks and the performance of the sequential models was assessed by evaluating each tuned model on 15 random samples across an increasing number of outcome variables. The average performance is displayed in Figure 4. The MTL models consistently outperformed the single-task representation, with the performance of both increasing with the number of tasks. The unweighted MTL models performed as well as or better than the uncertainty weighted MTL models. This result was contrary to expectations and led to an additional analysis exploring the properties of the uncertainty estimated loss weights.

Uncertainty Weighted Loss Weights Simulation

An additional investigation was conducted on the flexibility of the estimated loss weights using Kendall et al.'s (2018) uncertainty estimation. To better understand the similarity between the weighted and unweighted MTL model results, we examined the range of optimal loss weights for an individual task with varying levels of accuracy. We optimized the loss weight with respect to the uncertainty estimated binary cross-entropy for a single classification subcomponent of the overall multi-task framework (Figure 5). Results showed that the uncertainty estimation method provides limited flexibility for reweighting across the most common ranges of accuracy. The accuracy of the weighted MTL models ranged between 55-76% for each classification task, with loss weights between .77-1.07. This result is expected, as within this accuracy range there is a limited range of loss weights.

Table 1: Performance Comparison of Post-Test Sum across Static Baseline Models

Metric	Mean	GB	Lasso	Lin. SVR	RF	MLP	Majority Class MTL
MSE	13.91	9.77	**8.69**	12.14	8.76	12.49	18.22
MAE	3.19	2.54	**2.29**	2.81	2.41	2.85	3.49
R^2	-0.00	0.30	**0.37**	0.13	**0.37**	0.10	-.29

Table 2: Performance Comparison of Post-Test Sum across Neural Sequential Models

Metric	Single-task Model			Unweighted MTL			Weighted MTL		
Attention	None	Simple	Full	None	Simple	Full	None	Simple	Full
MSE	8.36	8.19	8.08	**6.93**	7.05	6.99	7.40	7.29	**6.92**
MAE	2.25	2.23	2.22	**2.06**	2.09	2.08	2.19	2.14	**2.07**
R^2	.41	.42	.42	**.51**	.50	.50	.47	.48	**.51**

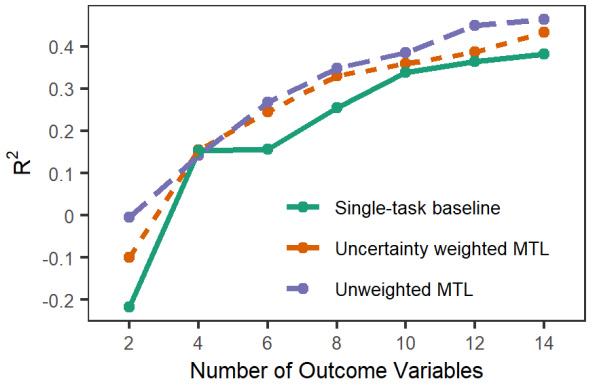

Figure 4: Sequential Model Performance by Number of Tasks.

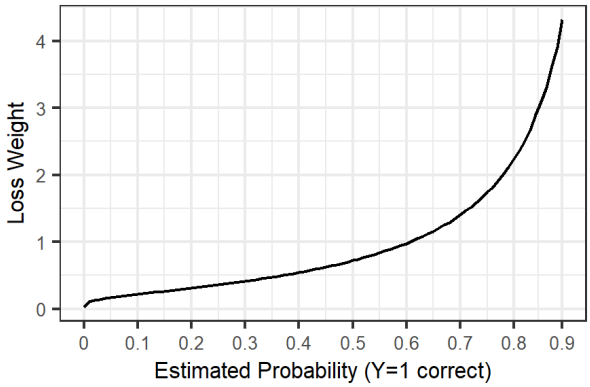

Figure 5: Optimal Uncertainty Estimated Loss Weight for a Single-Task.

Discussion

Evaluation results demonstrated that the multi-task learning (MTL) formulation of predictive student modeling yielded a 24% improvement in R^2 over the single-task neural network model using a sequential representation and a 38% improvement over models employing a static representation. Results showed that models leveraging the sequential nature of student interaction data outperformed those that used a static representation only.

Within the MTL framework, we observed an increase in model performance as the number of tasks increased across all models, with MTL models consistently outperforming the single-task model. Previous work on predictive student modeling in adaptive learning environments has typically reported R^2 ranges from 0.09 to 0.41, depending on the dataset and the chosen models (Moo, Lin, & Chi 2018; Bakker & Heskes 2003; Zhang et al. 2017). These results are in line with the model accuracies observed for the static baseline models utilized in this work. By leveraging a multi-task stacked LSTM framework, we observe sizable improvements in predictive accuracy.

In addition to the MTL framework, we used a self-attention mechanism to further act as a weighting scheme for modeling student's sequential gameplay data. We did not see substantial improvements from this self-attention mechanism. A potential explanation for this could be that each of the 17 tasks in the predictive modeling problem are influenced by different parts of the input sequence. Students are likely to gain knowledge throughout their interaction with the CRYSTAL ISLAND game-based learning environment. Therefore, predictions about the collection of tasks, each corresponding to a single item from the content knowledge post-test, may rely fully on the entire gameplay sequence. We constructed the attention mechanism as part of the shared weight portion of the model architecture, and it was an alternative to using attention for each unique task. This was due to insufficient data and the computational expense that task-specific attention would require. Because of this, attention may be forcing equal weighting across the game sequence because the tasks as a whole demand it.

It is notable that we did not see a benefit of using uncertainty weighting estimation over unweighted MTL models. Simulations on the uncertainty weighted loss weights shed light on this finding by demonstrating that the range of optimal loss weights is constrained when each of the tasks has a similar base rate, which is true in our dataset. These results suggest that when tasks in a multi-task framework possess similar base rates, the simpler method of equal weighting of tasks is as effective as more complex uncertainty-weighted methods.

Overall, results show that multi-task stacked LSTMs are an effective framework for predictive student modeling in educational games, and therefore, they show significant promise for enabling run-time support functionalities to enhance student learning in adaptive learning environments. Specifically, they enable personalized support, such as feedback and hints, that proactively intervene when a learner is trending toward a negative outcome. This support can also be targeted toward specific concepts and skills addressed by individual test items captured in the multi-task model. MTL is broadly applicable to predictive student modeling tasks, so long as they feature assessments with multiple questions, as is common in educational settings. Furthermore, MTL is likely to be most effective as the communality of test items decreases. Finally, predictive student models can also serve as a type of formative assessment, providing an "early warning system" for teachers that enables re-allocation of attention toward those students who need the most help.

Conclusion and Future Work

Predictive student modeling is critical for driving personalized feedback and support in adaptive learning environments. However, devising accurate models of student knowledge is challenging because student data for a particular learning environment may be sparse, and it is often inherently noisy. In this paper, we have introduced a multi-task stacked LSTM-based predictive student modeling framework for modeling student knowledge in educational games. Multi-task learning creates shared and task-specific representations of student learning data that improve model regularization and allow for increased flexibility in modeling different tasks.

In future work, it will be important to explore how different loss functions can be used to combine the loss across multiple correlated binary variables without requiring the assumption of independence across each task. It will also be important to investigate the performance of multi-task stacked LSTMs for predictive student modeling in different genres of learning environments to study their generalizability. Additionally, further research is needed on developing interpretable predictions for multi-task predictive student models to allow teachers to incorporate model feedback into classroom settings. Finally, it will be important to investigate the incorporation of the multi-task stacked LSTM-based predictive modeling framework in adaptive learning environments to explore how they can most effectively drive adaptive support to create the most effective learning experiences for students.

Acknowledgements

This material is based upon work supported by the National Science Foundation (NSF) under Grant DRL-1661202 and the Social Sciences and Humanities Research Council of Canada (SSHRC 895-2011-1006). Any opinions, findings, and conclusions expressed in this material are those of the authors and do not necessarily reflect the views of the NSF or SSHRC.

References

Argyriou, A.; Evgeniou, T.; and Pontil, M. 2007. Multi-Task Feature Learning. In Advances in Neural Information Processing Systems, 41–48.

Baker, R.; Corbett, A.; and Aleven, V. 2008. More Accurate Student Modeling Through Contextual Estimation of Slip and Guess Probabilities in Bayesian Knowledge Tracing. In Proceedings of the International Conference on Intelligent Tutoring Systems, 406-415. Berlin: Springer. doi.org/10.1007/978-3-540-69132-7_44

Bakker, B., and Heskes, T. 2003. Task Clustering and Gating for Bayesian Multitask Learning. *Journal of Machine Learning Research* 4(2003): 83–99.

Bosch, N.; D'Mello, S.; Baker, R.; Ocumpaugh, J.; Shute, V.; Wang, L.; and Zhao, W. 2016. Detecting Student Emotions in Computer-Enabled Classrooms. In Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence, 4125-4129. Palo Alto, CA: AAAI Press.

Corbett, A., and Anderson, J. 1994. Knowledge Tracing: Modeling the Acquisition of Procedural Knowledge. *User*

Modeling and User-Adapted Interaction, 4(4): 253–278. doi.org/10.1007/BF01099821

Education Superhighway. 2018. 2018 State of the States: Expanding Digital Learning to Every Classroom, Every Day. San Francisco, CA. Retrieved from https://s3-us-west-1.amazonaws.com/esh-sots-pdfs/2018%20State%20of%20the%20States.pdf

Embretson, S., and Reise, S. 2013. *Item Response Theory*. Psychology Press.

Fang, Y.; Ma, Z.; Zhang, Z.; Zhang, X.; and Bai, X. 2017. Dynamic Multi-Task Learning with Convolutional Neural Network. In Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, 1668–1674. Palo Alto, CA: AAAI Press. doi.org/10.24963/ijcai.2017/231

Gardner, J.; Brooks, C.; and Baker, R. 2019. Evaluating the Fairness of Predictive Student Models Through Slicing Analysis. In Proceedings of the Ninth International Learning Analytics and Knowledge Conference, 225–234. New York, NY: ACM. doi.org/10.1145/3303772.3303791.

Goncalves, A.; Von Zuben, F.; and Banerjee, A. 2016. Multi-task Sparse Structure Learning with Gaussian Copula Models. *Journal of Machine Learning Research 17*(1): 1205–1234.

Jin, B.; Yang, H.; Xiao, C.; Zhang, P.; Wei, X.; and Wang, F. 2017. Multitask Dyadic Prediction and Its Application in Prediction of Adverse Drug-Drug Interaction. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 1367–1373. Palo Alto, CA: AAAI Press.

Kendall, A.; Gal, Y.; and Cipolla, R. 2018. Multi-Task Learning Using Uncertainty to Weigh Losses for Scene Geometry and Semantics. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 7482–7491. IEEE Computer Society. doi.org/10.1109/CVPR.2018.00781

Khajah, M.; Huang, Y.; Gonzalez-Brenes, J.; Mozer, M.; and Brusilovsky, P. 2014. Incorporating Latent Factors into Knowledge Tracing to Predict Individual Differences in Learning. In Proceedings of the Seventh International Conference on Educational Data Mining, 99–106. International Educational Data Mining Society.

Lan, A. S.; Waters, A.; Studer, C.; and Baraniuk, R. 2014. Sparse Factor Analysis for Learning and Content Analytics. *Journal of Machine Learning Research*, *15*(1): 1959–2008.

Luong, M.; Pham, H.; and Manning, C. 2015. Effective Approaches to Attention-based Neural Machine Translation. arXiv preprint arXiv:1508.04025 [cs.CL]. Ithaca, NY: Cornell University Library.

Mao, Y.; Lin, C.; and Chi, M. 2018. Deep Learning vs. Bayesian Knowledge Tracing: Student Models for Interventions. *Journal of Educational Data Mining, 10*(2): 28–54.

Min, W.; Frankosky, M.; Mott, B.; Rowe, J.; Smith, A.; Wiebe, E.; Boyer, K.; and Lester, J. 2019. DeepStealth: Game-based Learning Stealth Assessment with Deep Neural Networks. *IEEE Transactions on Learning Technologies*. doi.org/10.1109/TLT.2019.2922356

Min, W.; Mott, B.; Rowe, J.; Taylor, R.; Wiebe, E.; Boyer, K.; and Lester, J. 2017. Multimodal Goal Recognition in Open-World Digital Games. In Proceedings of the Thirteenth Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 80–86. Palo Alto, CA: AAAI Press.

Mislevy, R., and Haertel, G. 2006. Implications of Evidence-Centered Design for Educational Testing. *Educational Measurement: Issues and Practice*, *25*(4): 6–20. doi.org/10.1111/j.1745-3992.2006.00075.x

Pardos, Z., and Heffernan, N. 2011. KT-IDEM: Introducing Item Difficulty to the Knowledge Tracing Model. In Proceedings of the Nineteenth International Conference on User Modeling, Adaptation, and Personalization, 243–254. Berlin: Springer. doi.org/10.1007/978-3-642-22362-4_21.

Piech, C.; Bassen, J.; Huang, J.; Ganguli, S.; Sahami, M.; Guibas, L.; and Sohl-Dickstein, J. 2015. Deep Knowledge Tracing. In Advances in Neural Information Processing Systems, 505–513.

Rowe, J.; Shores, L.; Mott, B.; and Lester, J. 2011. Integrating Learning, Problem Solving, and Engagement in Narrative-Centered Learning Environments. *International Journal of Artificial Intelligence in Education*, 21(1-2): 115–133.

Sawyer, R.; Rowe, J.; Azevedo, R.; and Lester, J. 2018. Modeling Player Engagement with Bayesian Hierarchical Models. In Proceedings of the Fourteenth Artificial Intelligence and Interactive Digital Entertainment Conference.

Shui, C.; Abbasi, M.; Robitaille, L.; Wang, B.; and Gagne, C. 2019. A Principled Approach for Learning Similarity in Multitask Learning. In Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, 3446–3452. Palo Alto, CA: AAAI Press. doi.org/10.24963/ijcai.2019/478.

Shute, V.; Wang, L.; Greiff, S.; Zhao, W.; and Moore, G. 2016. Measuring problem solving skills via stealth assessment in an engaging video game. *Computers in Human Behavior* 63: 106–117. doi.org/10.1016/j.chb.2016.05.047

Su, Y.; Liu, Q.; Liu, Q.; Huang, Z.; Yin, Y.; Chen, E.; Ding, C.; Wei, S.; and Hu, G. 2018. Exercise-Enhanced Sequential Modeling for Student Performance Prediction. In Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, 2435–2443. Palo Alto, CA: AAAI Press.

Tong, X.; Fu, Z.; Shang, M.; Zhao, D.; and Yan., R. 2018. One "Ruler" for All Languages: Multi-Lingual Dialogue Evaluation with Adversarial Multi-Task Learning. In Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, 4432–4438. Palo Alto, CA: AAAI Press. doi: doi.org/10.24963/ijcai.2018/616.

U.S. Department of Education 2016. Future Ready Learning: Reimagining the Role of Technology in Education. Washington, DC: Office of Educational Technology. https://tech.ed.gov/files/2015/12/NETP16.pdf.

VanLehn, K. 2011. The Relative Effectiveness of Human Tutoring, Intelligent Tutoring Systems and Other Tutoring Systems. *Educational Psychologist*, 46(4): 197–221. doi.org/10.1080/00461520.2011.611369.

Zhang, L.; Xiong, X.; Zhao, S.; Botelho, A.; and Heffernan, N. T. 2017. Incorporating Rich Features into Deep Knowledge Tracing. In Proceedings of the Fourth ACM Conference on Learning @ Scale, 169–172. New York, NY: ACM. doi.org/10.475/123_4.

Zhang, Y., and Yang, Q. 2017. Learning Sparse Task Relations in Multi-Task Learning. In Proceedings of the Thirty-First AAAI Conference on Artificial Intelligence, 2914–2920. Palo Alto, CA: AAAI Press.