Applying Representation Learning for Educational Data Mining

Milagro Teruel
Universidad Nacional de Córdoba
Córdoba, Argentina
milagro.teruel@gmail.com

Laura Alonso Alemany
Universidad Nacional de Córdoba
Córdoba, Argentina
lauraalonsoalemany@gmail.com

ABSTRACT

Educational Data Mining is an area of growing interest, given the increase of available data and generalization of online learning environments. In this paper we present a first approach to integrating Representation Learning techniques in Educational Data Mining by adding autoencoders as a preprocessing step in a standard performance prediction problem.

Preliminary results do not show an improvement in performance by using autoencoders, but we expect that a fine tuning of parameters will provide an improvement. Also, we expect that autoencoders will be more useful combined with different kinds of classifiers, like multilayer perceptrons.

1. INTRODUCTION

The area of Educational Data Mining (EDM) has grown rapidly in the last years, given the popularization of Massive Open Online Courses (MOOCs) like in Coursera or EdX, and the use of tutoring systems. In this area there are a varied set of challenges to adapt traditional teaching models to new environments and scales, like in the case of online learning. Moreover, there is a growing interest to integrate into the learning process the information provided by these new educational platforms. Data is collected automatically, usually in big amounts of low level activity logs. Integrating this data is a challenge because the amount of data and its low level of abstraction hinder its interpretation and posterior application [8].

Some of the most popular tasks in EDM are dropout prediction [5], student modeling and knowledge tracing [10] [7], prediction of student performance [13] [6], and modeling of forum activity [3].

To the extent of our knowledge, many of the most successful approaches to these tasks [6] [8] [3] rely on the use of hand crafted features constructed aggregating lower level information, such as the total number of visits to a particular page or average performance on a given exercise. Some approaches resort to latent concepts that are also hand-crafted [9]. This process adds prior knowledge from experts to the raw log data as bias to guide the model to select decision criteria. However, this method has some disadvantages: the features extracted from a course may not be applicable to different platforms, for example, knowledge inferred from a hierarchical structure particular to a given course cannot be extrapolated to a course with a different structure. Moreover, the design of the features can be labor-intensive and require the help of an expert. Both these problems are discussed in [15] and [2].

Our long-term goal is to apply Representation Learning techniques [1] to automatically discover factors of variation in data and render them interpretable for consumption of instructional designers. An efficient representation can evidence factors that explain the latent causes of data distribution, bringing a better understanding of why students behave as they do in the courses.

As a first approach to this problem, we have applied the most fundamental Representation Learning techniques to a standard Educational Data Mining problem, and we have analyzed the results to guide future work. We found this task has some challenging aspects where Representation Learning can prove specially useful to discover latent factors. On the one side, the data is highly sparse because not all students have attempted all problems. A more dense representation can highlight patterns more easily. On the other side, there is a strong temporal component of the problem, given that students improve during time if they practice, or forget concepts if they don't.

2. PREDICTION OF STUDENT PERFORMANCE ON MATHEMATICAL PROBLEMS

As a testbed for different approaches to Representation Learning in EDM, we have used the challenge proposed in the KDDCup 2010 [12].

2.1 Task description

The task for KDDCup challenge 2010 consisted in predicting if a student is going to correctly solve a problem for the first time, based on the records of all previous interactions with the tutoring system. The expected predictions are real numbers between 0 and 1, and the evaluation metric was the root mean squared error (RMSE).

Two datasets, from two different tutoring systems but with the same structure, were released for development: *Algebra 2008-2009* and *Bridge to Algebra 2008-2009*. They sum up to 29 million attempts of 9,000 students solving

problems. Each row includes identification for student and problem, time and duration of interaction, the position of the problem in the course hierarchy and Knowledge Components related to the problem. Additionally, the number of times the students has previously seen the problem or the Knowledge Component is also available in data. In the test dataset, columns with temporal information have been purposely deleted because they are highly correlated with the target to predict.

In our experimental setting, for each experiment, we obtained results with 5 samples of 500,000 records of the Algebra part of the dataset.

2.2 Previous work

The winning team in the competition [6] used a large ensemble of classifier and aggregated features designed by six independent teams.

Other very interesting approach that won the third place [14] uses a model similar to traditional collaborative filtering, incorporating Matrix Factorization. The students are represented as users and the problems are represented as items, thus predicting if a student will resolve correctly a problem can be seen as predicting if a user will be interested in a given item. Matrix Factorization is a Representation Learning method, but we think Autoencoders may provide a more adequate representation because they are more general models.

The fourth place was won by [9], with an approach of modeling students and learned skills (or knowledge components in this case) as a Hidden Markov Model.

The SPARFA approach [7] induces the states of a Markov Model to model student's behaviour, like the one used in [9]. [10] also induces a graph-like representation of student knowledge using Long-Short Term Memory Networks.

2.3 Experimental Setting

We assess the impact of preprocessing data with Autoencoders [11] in the classification performance. We compare this approach with the most popular approach to the problem, Matrix Factorization [14] [13], described above and implemented as in [17], which can be considered the state-of-the-art for this task. Since we did not have access to the challenge evaluation dataset, we built our own following the method in `https://pslcdatashop.web.cmu.edu/KDDCup/rules_data_format.jsp`.

As base classifiers, we used Naïve Bayes and Logistic Regression (with a regularization parameter of the L2 norm of 0.01 and a sigmoid activation). For our first approach, we implemented autoencoders using the Keras library [4]. In these first experiments, we used adam optimizer and representation sizes of 50 to 500. We also evaluated denoising autoencoders [16], with a noise factor of 0.1.

The baseline for this task, the average performance of all students by problem, reaches a 0.33 RMSE. Matrix factorization introduces a slight improvement, obtaining 0.31 RMSE. Logistic Regression without autoencoders obtains 0.33, while adding an autoencoder or denoising autoencoder as a preprocess produces a decrease in performance, ranging from 0.34 to 0.54, depending on representation size, the best size being 100. In the case of Naïve Bayes, the performance of the classifier alone is 0.39 RMSE, but in this case adding an autoencoder does produce an increase in performance, achieving 0.34.

3. CONCLUSIONS AND FUTURE WORK

We have presented a first approach to integrating Representation Learning techniques in Educational Data Mining by adding autoencoders as a preprocessing step in a standard performance prediction problem.

The analysis of preliminary results shows that using autoencoders as a preprocess without a fine-tuning of parameters does not improve performance of a Logistic Regression classifier. It does improve the performance of a Naïve Bayes classifier, but that does not reach the baseline performance.

We expect that autoencoders will be more useful for classifiers that are able to shape more complex decision boundaries, like multilayer perceptrons. We are also exploring different configurations of parameters and expect to gain some improvement in performance there.

In future work we will be using other projection methods as a preprocess. We are also interested in incorporating hand-crafted features and assess their impact in the resulting projections. We are particularly interested in the interpretability of the obtained projection spaces, and hope that hand-crafted features may be a good starting point.

4. REFERENCES

[1] Y. Bengio, A. C. Courville, and P. Vincent. Unsupervised feature learning and deep learning: A review and new perspectives. *CoRR*, abs/1206.5538, 2012.

[2] S. Boyer and K. Veeramachaneni. *Artificial Intelligence in Education: 17th International Conference, AIED 2015, Madrid, Spain, June 22-26, 2015. Proceedings*, chapter Transfer Learning for Predictive Models in Massive Open Online Courses, pages 54–63. Springer International Publishing, Cham, 2015.

[3] S. Chaturvedi, D. Goldwasser, and H. Daumé III. Predicting instructor's intervention in mooc forums. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2014.

[4] F. Chollet. Keras. `https://github.com/fchollet/keras`, 2015.

[5] G. W. Dekker, M. Pechenizkiy, and J. M. Vleeshouwers. Predicting students drop out: A case study. *International Working Group on Educational Data Mining*, 2009.

[6] H. fu Yu, H. yi Lo, H. ping Hsieh, J. kai Lou, T. G. Mckenzie, J. wei Chou, P. han Chung, C. hua Ho, C. fu Chang, J. yu Weng, E. syu Yan, C. wei Chang, T. ting Kuo, P. T. Chang, C. Po, C. yuan Wang, Y. hung Huang, Y. xun Ruan, Y. shi Lin, S. de Lin, H. tien Lin, and C. jen Lin. Feature engineering and classifier ensemble for kdd cup 2010. In *In JMLR Workshop and Conference Proceedings*, 2011.

[7] A. S. Lan, A. E. Waters, C. Studer, and R. G. Baraniuk. Sparse factor analysis for learning and content analytics. *J. Mach. Learn. Res.*, 15(1):1959–2008, Jan. 2014.

[8] U.-M. O'Reilly and K. Veeramachaneni. Technology for mining the big data of moocs. *Research & Practice in Assessment*, 9(2):29–37, 2014.

[9] Z. A. Pardos and N. T. Heffernan. 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 W & CP*, 2001.

[10] C. Piech, J. Bassen, J. Huang, S. Ganguli, M. Sahami, L. J. Guibas, and J. Sohl-Dickstein. Deep knowledge tracing. In *Advances in Neural Information Processing Systems*, pages 505–513, 2015.

[11] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Neurocomputing: Foundations of research. chapter Learning Representations by Back-propagating Errors, pages 696–699. MIT Press, Cambridge, MA, USA, 1988.

[12] R. S. G. G. Stamper J., Niculescu-Mizil A. and K. Koedinger. Algebra l 2008-2009 and bridge to algebra 2008-2009. challenge data set from kdd cup 2010 educational data mining challenge. find it at http://pslcdatashop.web.cmu.edu/kddcup/downloads.jsp. 2010.

[13] N. Thai-nghe, L. Drumond, A. Krohn-grimberghe, R. Nanopoulos, and L. Schmidt-thieme. Factorization techniques for predicting student performance. In *In Educational Recommender Systems and Technologies: Practices and Challenges (In*, 2011.

[14] A. Toscher and M. Jahrer. Collaborative filtering applied to educational data mining. *KDD cup*, 2010.

[15] K. Veeramachaneni, U. O'Reilly, and C. Taylor. Towards feature engineering at scale for data from massive open online courses. *CoRR*, abs/1407.5238, 2014.

[16] P. Vincent, H. Larochelle, Y. Bengio, and P.-A. Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th International Conference on Machine Learning*, ICML '08, pages 1096–1103, New York, NY, USA, 2008. ACM.

[17] H.-F. Yu, C.-J. Hsieh, S. Si, and I. Dhillon. Scalable coordinate descent approaches to parallel matrix factorization for recommender systems. In *Proceedings of the 2012 IEEE 12th International Conference on Data Mining*, ICDM '12, pages 765–774, Washington, DC, USA, 2012. IEEE Computer Society.