Peer Assessment in MOOCs Using Preference Learning via Matrix Factorization

Jorge Díez
Artificial Intelligence Center
University of Oviedo – 33204 Gijón (Spain)
jdiez@uniovi.es

Oscar Luaces
Artificial Intelligence Center
University of Oviedo – 33204 Gijón (Spain)
oluaces@uniovi.es

Amparo Alonso-Betanzos
Department of Computer Science
Faculty of Informatics, University of A Coruña
15071 A Coruña, Spain
ciamparo@udc.es

Alicia Troncoso
Department of Computer Science
Pablo de Olavide University of Seville
41013 Sevilla, Spain
atrolor@upo.es

Antonio Bahamonde
Artificial Intelligence Center
University of Oviedo – 33204 Gijón (Spain)
abahamonde@uniovi.es

Abstract

Evaluating in Massive Open Online Courses (MOOCs) is a difficult task because of the huge number of students involved in the courses. Peer grading is an effective method to cope with this problem, but something must be done to lessen the effect of the subjective evaluation. In this paper we present a matrix factorization approach able to learn from the order of the subset of exams evaluated by each grader. We tested this method on a data set provided by a real peer review process. By using a tailored graphical representation, the induced model could also allow the detection of peculiarities in the peer review process.

1 Introduction

In the last few years, Massive Open Online Courses (MOOCs) have increased their popularity. These courses make University lectures available to tens of thousands of students at a time. MOOCs can be suitable even for highly experimental subjects, where students can make real experiments with simulated materials (microscopes, etc) or can take real data from remote controlled devices [9].

However, despite the technical challenges inherent to the deployment of a MOOC, there is an important difficulty that has to be addressed: it is necessary to evaluate a very large number of exams that cannot be automatically evaluated, like open-ended exercises or essays. A practical approach to cope with this problem is known as *peer grading* or *peer assessment* [8], in which students grade a small amount of exams

submitted by other students. These graders are provided with a rubric which specifies the evaluation criteria to be followed to assess the questions raised in the exam. The final score given to an exam is usually determined as the average (or median) of the corresponding peer-grades [6] given by the evaluators.

This naive approach has some drawbacks. Firstly, we have to expect that assessments will be affected by some graders' bias that would deviate them with respect to the *ideal ground truth*. To compensate such deviation, it is essential for each exam to be evaluated by many graders so that the correct marks will be approximated by their average assessments. Moreover, assuming that each exam was graded by a big amount of students, it has been reported [7] that averages are more consistently accurate with respect to the rubric than the staff grades. But unfortunately, we only can handle a reduced number of assessments for each exam.

Secondly, there is a kind of *batch effect* in evaluation tasks prone to subjectivity. It has been observed [1, 3, 5] that an item tends to receive a higher score when it is evaluated in a batch of worse items than when it is evaluated in a group of better items. Fortunately, it has also been observed that, despite the graders' biases, the ranking entailed by their assessments is coherent with the ground truth; i.e., the scores can be unreliable but the order is, in general, correctly assessed [1, 3, 8].

In this paper we present a new factorization method [10, 11] that learns from preference judgments [4, 5]. The method takes advantage of the coherence in the ranking given by the graders, while avoiding the drawbacks mentioned above. Our approach predicts a ranking of exams that can be easily translated to scores. In addition to that, exams and graders (students) are mapped to an Euclidean space where it will be easy to apply post-processing techniques like clustering. In the case that the mapping space has 3 or less dimensions, visual analysis can even be achieved using the graphical representation that will be presented in the results section.

In the rest of the paper we make a formal presentation of our approach and we present some experimental results on real-world data.

2 Formal Framework

Let us consider the following dataset

$$D \subset Graders \times Exams \times Scores, \tag{1}$$

where each element is of the form $(\boldsymbol{g}, \boldsymbol{e}, f(\boldsymbol{g}, \boldsymbol{e}))$, being $f(\boldsymbol{g}, \boldsymbol{e}) \in \mathbb{R}$ the score given by $\boldsymbol{g} \in Graders$ to $\boldsymbol{e} \in Exams$. Starting from D, we can build a dataset of *preference judgments*

$$D_{pj} = \{(\boldsymbol{g}, \boldsymbol{e}, \boldsymbol{e'}) : (\boldsymbol{g}, \boldsymbol{e}, f(\boldsymbol{g}, \boldsymbol{e})), (\boldsymbol{g}, \boldsymbol{e'}, f(\boldsymbol{g}, \boldsymbol{e'})) \in D, f(\boldsymbol{g}, \boldsymbol{e}) > f(\boldsymbol{g}, \boldsymbol{e'})\} \tag{2}$$

thus discarding the numerical scores but preserving the information of the ranking, since the triplet $(\boldsymbol{g}, \boldsymbol{e}, \boldsymbol{e'})$ indicates that $\boldsymbol{g}$ rates $\boldsymbol{e}$ higher than $\boldsymbol{e'}$.

The aim is to find an *utility* function ut that depends not only on the input data (grader and exam) but also on some additional parameters θ, such that the variations of f can be predicted by the variations of g in the sense that

$$f(\boldsymbol{g}, \boldsymbol{e}) > f(\boldsymbol{g}, \boldsymbol{e'}) \Longleftrightarrow ut(\boldsymbol{g}, \boldsymbol{e}, \theta) > ut(\boldsymbol{g}, \boldsymbol{e'}, \theta). \tag{3}$$

We want to determine the optimal values, θ^*, those that minimize a given loss function, $\mathcal{L}$, plus a Gaussian regularization:

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \sum_{(i,j) \in D_{pj}} \mathcal{L} + \nu \mathcal{R}(\theta). \tag{4}$$

This equation will be solved using an Stochastic Gradient Descent (SGD) algorithm.

Following a maximum margin approach, we will use the following loss function

$$\mathcal{L} = L_{AUC}(D_{pj}) = \sum_{(\boldsymbol{g}_i, \boldsymbol{e}_i, \boldsymbol{e'}_i) \in D_{pj}} \max\Big(0, 1 - ut(\boldsymbol{g}_i, \boldsymbol{e}_i, \theta) + ut(\boldsymbol{g}_i, \boldsymbol{e'}_i, \theta)\Big). \tag{5}$$

The idea is to ensure that the difference of the utilities in a preference judgment is at least 1.

Input data can be given just by unique identifiers for graders and exams. However, we could have additional descriptors including additional information. For example, each exam may have attached the scores of the same student in previous exams. Nevertheless, in [7] the authors do not recommend these additional information, since it prevents students to be evaluated from a *clean state*.

In any case, the identifiers of exams and students (our graders) can be binarized, and without loss of generality, we can assume that *graders* are described by vectors in an Euclidean space of dimension G, and *exams* are given by vectors with E components. We shall use two linear maps, given respectively by matrices $\boldsymbol{W}$ and $\boldsymbol{V}$, to represent them in the same k-dimensional space:

$$\mathbb{R}^G \longrightarrow \mathbb{R}^k,\ \boldsymbol{g} \rightsquigarrow \boldsymbol{W}\boldsymbol{g} \qquad \text{and} \qquad \mathbb{R}^E \longrightarrow \mathbb{R}^k,\ \boldsymbol{e} \rightsquigarrow \boldsymbol{V}\boldsymbol{e}. \tag{6}$$

Therefore, function g can be defined as the interaction between the grader and the exam in terms of Euclidean distance in $\mathbb{R}^k$:

$$ut(\boldsymbol{g}, \boldsymbol{e}, \theta) = ut(\boldsymbol{g}, \boldsymbol{e}, \boldsymbol{W}, \boldsymbol{V}) = -\left\|\boldsymbol{W}\boldsymbol{g} - \boldsymbol{V}\boldsymbol{e}\right\|^2 \tag{7}$$

We need to add a constant term (usually it takes value 1) to each input description to guarantee that the utility function includes all monomials up to degree 2 formed with variables taken from the description of graders (g) and exams (e). On the other hand, to implement regularization we use the square of the Frobenius norm.

Once we have learned the utility function ut, the final score of an exam e is defined as the average of the predicted scores given by *all* the graders,

$$\text{score}(e) = \frac{\sum_{g \in Graders} ut(\boldsymbol{g}, \boldsymbol{e}, \boldsymbol{W}, \boldsymbol{V})}{|Graders|} = -\frac{\sum_{g \in Graders} \left\|\boldsymbol{W}\boldsymbol{g} - \boldsymbol{V}\boldsymbol{e}\right\|^2}{|Graders|}, \tag{8}$$

This score will be used to make the final ranking. The ranking can be *calibrated* to transform percentiles into valid grades. This can be done using a table of equivalences or using marks provided by the staff for some exams to make an interpolation.

3 Experimental validation

There is a work in progress to peer review the exams in a course of Intelligent Systems with about 300 students belonging to three different universities in Spain (University of A Coruña, University Pablo de Olavide at Sevilla, and University of Oviedo at Gijón). We plan to apply the method detailed above to these peer assessments. Unfortunately, these data will not be available until mid-2014. Therefore, we carried out our experiments with a data set provided by the review process of the CAEPIA '13 conference, a biennial congress promoted by the Artificial Intelligence Spanish Association.

In this context, reviewers can be considered as students grading exams (papers) of other students. Any paper review process is characterized by the lack of a true solution because there are no fixed questions for all the papers to be answered. Moreover, each paper can address a different topic. However, there is a general rubric that all reviewers keep in mind (originality, writing quality, scientific soundness, etc...) to assess the papers regardless of the topic. Thus, they can be considered equivalent to students' exams for the purpose of evaluating a grading method.

There were 98 papers submitted to the different tracks of the conference; each one was reviewed by no less than 3 reviewers, who had to rate the papers in the range 1 (*strong reject*) to 5 (*strong accept*). They were also asked to indicate their confidence in the evaluation from 1 (*none*) to 5 (*expert*). Then, a weighted score for each pair {reviewer, paper} was computed as $rate \times confidence$. Finally, the weighted scores were averaged to obtain the final score for each paper, and, therefore, the ranking that allows to make the decision of acceptance or rejection. Both the identifiers of papers and reviewers were changed to preserve anonymity.

Note that each track of the conference had its own set of reviewers and papers. Figure (1) depicts the graphs of two different tracks, where reviewers are connected to the papers they reviewed. Each track can be represented as a graph, which is isolated from any other track's graph, since neither reviewers nor papers are shared among tracks. On the other hand, it is important to have a connected graph for each track in order to obtain a reliable ranking.

We built a data set of 271 preference judgments from the scores given by the reviewers as described in (2). Following the advice given in [7], reviewers and papers were described only by a unique identifier, avoiding the use of other available features that could bias the grades.

3.1 Model evaluation

The first experiment that we carried out was devised to estimate the quality of the models learned by our factorization approach. We cannot use a cross-validation for this purpose, since there could be examples in the test fold where either the reviewer or the paper were missing in the training set, that is, there could be unknown identifiers in the test set. The mapping of these unknown items would be dependent on their arbitrary representation. Recall that our data set is made of triplets (g, e, e'), so we need a test set with triplets not included in the training set, but formed with three items that appeared somewhere (but not together) in the training set, so that the learned mapping makes sense for them.

Therefore, we made the following train-test experiment: for each reviewer with more than one preference judgement (triplet), which implies more than two papers reviewed, we randomly chose one of the triplets for the test set, and the rest were added to the training set. Reviewers who evaluated just two papers provided a single preference judgement that was used only in the training set. The result of this procedure yielded 216 and 55 preference judgments for the training and testing sets, respectively.

Since we are interested in the graphical capabilities of the method, we fixed the dimension of the mapping space to $k = 2$ and we searched for the best combination of values for the rest of parameters. The values that produced the best results yielded an AUC in the test set of 83.64%, that is, less than 1 out of 5 preference judgments (i.e. pairs of papers) were ranked in the wrong order. This result confirmed the reliability of the induced model.

3.2 Estimating the papers' scores

Once we have found a good combination of parameters to induce a reliable model, we used all the preference judgments to learn a new model. The utility function so learned can be used to infer the rates of all the reviewers for all the papers in each track. Recall that a higher number of grades used to compute the final score of a paper is expected to yield scores more consistently accurate with respect to the rubric [7]. Thus, the final score of a paper p will be computed using (8).

We tested our approach in the two tracks with a higher number of submissions in the conference, *Machine Learning* and *Artificial Intelligence Applications*. The AUC of the rankings obtained using (8) with respect to the final decision in the conference was 86.54% and 85.71%, respectively. That is, our model produced a consensus ranking of all reviewers that would change some final decisions in the conference or at least the order of some papers (see Table 1).

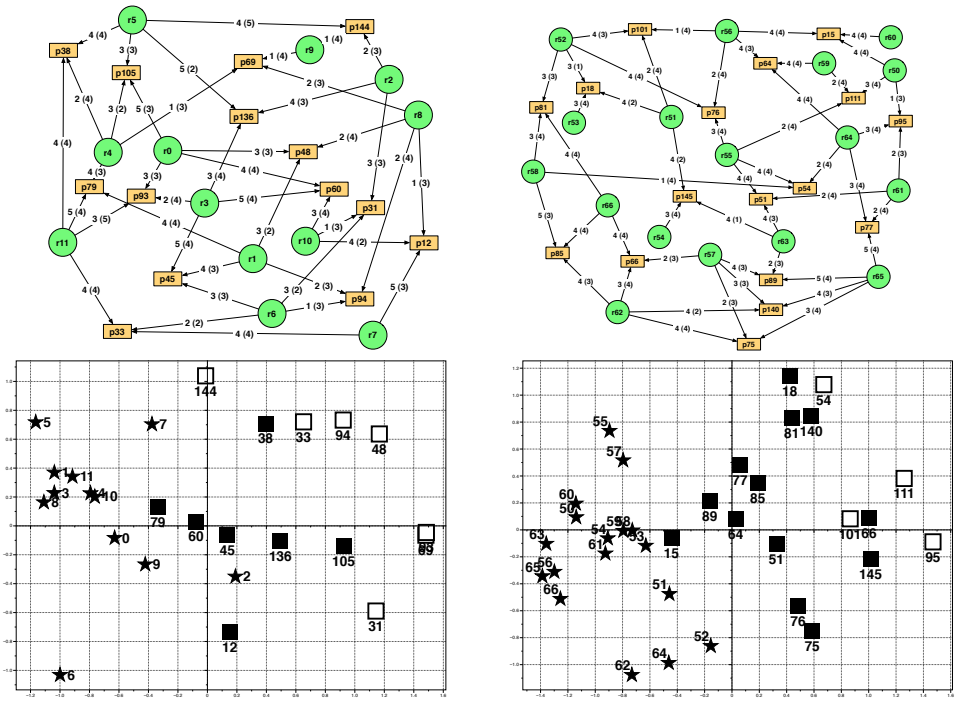

Figure 1: Papers and reviewers for tracks *AI Applications* (left) and *Machine Learning* (right). Numbers are fictitious to preserve the identity of the reviewers/authors. Subfigures in the upper part show graphs with arcs connecting reviewers (circles) to the papers (rectangles) reviewed by them, labeled with the rate and the reviewer's confidence. Graphs below show the mapping of reviewers (stars), accepted (filled squares) and rejected papers (hollow squares) induced by our factorization method.

3.3 Graphical representation

We can take advantage of the graphical representation of the Euclidean space where the items (reviewers and papers) are mapped, in order to gain some insight into the problem. We used a 2-dimensional space to represent the reviewers (stars) and papers (squares), as can be seen in Figure 1. Papers that were rejected in the conference were drawn as hollow squares. In general, the algorithm has mapped rejected papers further away from reviewers than accepted papers, as expected. However, a visual inspection reveals that there are some papers that could deserve a deeper analysis because they were rejected despite our model maps them closer to the reviewers than other accepted papers. That was the case for paper 144 in the *AI Applications* track (graph on the left) and for paper 101 in *Machine Learning* (graph on the right).

Table 1 shows the ranking of papers given by the reviewers as well as the ranking produced by our method. The papers rejected in the conference are shown in gray cells. As in the graphical representation, papers 144 and 101 clearly show up as the biggest discrepancies between both rankings.

4 Conclusions

We have devised a factorization method to implement peer assessment. Our approach learns from preference judgments to avoid the subjectivity of the numeric grades. In fact, our method satisfies the desiderata of an

		← Higher score									Lower score →							
AI	Conference	12	79	45	60	136	105	38	33	93	48	144	31	94	69			
	Model	79	60	45	144	12	136	38	33	105	94	48	31	69	93			
ML	Conference	89	85	15	64	140	145	81	77	18	51	66	75	76	54	111	101	95
	Model	15	89	64	77	85	51	76	75	81	101	140	18	145	66	54	111	95

Table 1: Ranking of the papers for the tracks *AI Aplications* (AI) and *Machine Learning* (ML) given by the reviewers and by our model. Numbers are fictitious to preserve the identity of the reviewers/authors. Cells in gray indicate papers rejected by the program committee.

ideal peer grading system for a MOOC [7]: i) it provides highly reliable/accurate assessment, ii) it allocates balanced and limited workload across students and course staff, iii) it is very fast (SGD-based) so it is easily scalable to a large number of students and iv) it can be applied to a diverse collection of problem settings.

We have tested it on a data set obtained from a reviewing process from the Spanish Conference on Artificial Intelligence (CAEPIA). The ranking obtained by our method shows a high coherence with the reviewers' ranking, although some differences showed up. Since our ranking is obtained with an accurate model and a higher number of grades, it is expected to be very reliable, so the differences with the reviewers' ranking could be pointing out a deficiency in the reviewers' assessment.

Acknowledgments

The research reported here is supported in part under grants TIN2011-23558, TIN2012-37954 and TIN2011-28956-C02 from the MINECO (Ministerio de Economía y Competitividad, Spain), all partially supported with FEDER funds. We would also like to thank Concha Bielza (President of the Program Committee of CAEPIA'13) for providing the data used in the experiments reported in the paper.

References

[1] Bahamonde, A., Bayón, G. F., Díez, J., Quevedo, J. R., Luaces, O., del Coz, J. J., Alonso, J., & Goyache, F. (2004). Feature subset selection for learning preferences: A case study. In *Procs. of the ICML '04*.

[2] Cohen, W. W., Schapire, R. E., & Singer, Y. (1999). Learning to order things. *Journal of Artificial Intelligence Research*, 10, 243–270.

[3] Díez, J., Bayón, G. F., Quevedo, J. R., del Coz, J. J., Luaces, O., Alonso, J., & Bahamonde, A. (2004). Discovering relevancies in very difficult regression problems: applications to sensory data analysis. In *Procs. of the ECAI '04*.

[4] Herbrich, R., Graepel, T., & Obermayer, K. (1999). Support vector learning for ordinal regression. In *Proceedings of the Ninth International Conference on Artificial Neural Networks* (pp. 97–102). Edinburgh, UK.

[5] Joachims, T. (2002). Optimizing search engines using clickthrough data. In *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD)*.

[6] Kulkarni, C., Pang-Wei, K., Le, H., Chia, D., Papadopoulos, K., Cheng, J., Koller, D., & Klemmer, S. R. (2013). *Peer and Self Assessment in Massive Online Classes*. Technical report, Stanford University.

[7] Piech, C., Huang, J., Chen, Z., Do, C., Ng, A., & Koller, D. (2013). Tuned models of peer assessment in MOOCs. In *Proceedings of the 6^{th} International Conference on Educational Data Mining (EDM'13)*.

[8] Sadler, P. M. & Good, E. (2006). The impact of self-and peer-grading on student learning. *Educational assessment*, 11(1), 1–31.

[9] Waldrop, M. M. (2013). Education online: The virtual lab. *Nature*, 499(268–270).

[10] Weston, J., Bengio, S., & Hamel, P. (2011). Multi-tasking with joint semantic spaces for large-scale music annotation and retrieval. *Journal of New Music Research*, 40(4), 337–348.

[11] Weston, J., Bengio, S., & Usunier, N. (2010). Large scale image annotation: Learning to rank with joint word-image embeddings. *Machine learning*, 81(1), 21–35.