Towards a General Purpose Anomaly Detection Method to Identify Cheaters in Massive Open Online Courses

Giora Alexandron[1], José A. Ruipérez-Valiente[2] and David E. Pritchard[2]
[1] Weizmann Institute of Science, Herzl St 234, Rehovot, Israel
[2] Massachusetts Institute of Technology, 77 Massachusetts Ave, Cambridge (MA), 02139, USA
giora.alexandron@weizmann.ac.il, jruipere@mit.edu, dpritch@mit.edu

ABSTRACT

We propose a general-purpose method for detecting cheating in Massive Open Online Courses (MOOCs) using an Anomaly Detection technique. Using features that are based on measures of aberrant behavior, we show that a classifier that is trained on data of one type of cheating (Copying Using Multiple Accounts) can detect users who perform another type of cheating (unauthorized collaboration). The study exploits the fact that we have dedicated algorithms for detecting these two methods of cheating, which are used as reference models. The contribution of this paper is twofold. First, we demonstrate that a detection method that is based on anomaly detection, which is trained on a known set of cheaters, can generalize to detect cheaters who use other methods. Second, we propose a new time-based person-fit aberrant behavior statistic.

Keywords

MOOCs; Learning Analytics; Anomaly Detection; Cheating

1. INTRODUCTION

Academic dishonesty is one of the endemic problems of higher education. Some studies have reported that up to 95% of college students are engaged in some kind of dishonest behavior [10, 4, 9]. The anonymity of online environments makes it easier for students to cheat [8]. In addition, online learning environments tend to be more heterogeneous, and students might differ significantly in their perception of what constitutes cheating [3]. These might lead students to look for ways to exploit the properties of a learning environment to gain credit without learning the contents [2, 3].

Within the context of online learning, Massive Open Online Courses (MOOCs) have greatly garnered the attention of the media and researchers over the last decade [5]. Estimates are that during 2018 there were more than 100M MOOC learners, more than 11k MOOCs, and more than 900 institutes involved [18]. These large numbers tell some of the story of how MOOCs change the educational landscape.

PRE-PRINT VERSION
Cite as Giora Alexandron, José A. Ruipérez-Valiente, and David E. Pritchard. 2019. Towards a General Purpose Anomaly Detection Method to Identify Cheaters in Massive Open Online Courses. In *Proceedings of the 12th International Conference on Educational Data Mining*. 480-483.

MOOCs offer certificates that do not have formal academic status, but are still perceived as valuable, for example in the labor market. Thus, it is not surprising that several studies reported on cheating in MOOCS, for example by plagiarizing peer-review assignments [7]. One of the main cheating methods that was discovered in MOOCs is Copying Using Multiple Accounts (CUMA). Several studies have been successful at detecting CUMA by implementing probabilistic algorithms [13], heuristics [2], and machine learning [17]. A different cheating method, unauthorized collaboration, was reported in [16], and detected using a method that is based on proximity of submissions in time. We note that CUMA and unauthorized collaboration are strictly forbidden by all the major MOOC platforms, e.g., edX [1].

With major MOOC providers currently pivoting towards the direction of professional development and online degrees [15], their is an even greater need for robust techniques to prevent and detect cheating, which can generalize and scale across platforms, topics and courses. The goal of this study is to develop general detection techniques that can identify cheating without assuming a specific pattern. The rationale is to rely on behavioral patterns that capture various types of aberrant behavior, rather than relying on temporal patterns that are specific to a certain method of cheating.

In previous work [1] we *hypothesized* that anomaly detection can be used to build such a general purpose classifier, which 'bootstrap' from one type of cheating to detect other types of cheating. However we were unable to demonstrate this, due to lack of a reference model. The current study extends [1], and demonstrates that a general-purpose cheater detector that is based on anomaly detection can be trained on one type of cheating and then be used to discover other type of cheating. The detector is based on measures of aberrant behavior, such as Guttman Error [11]. As an additional contribution, we also formalize a new time-based aberrant behavior person-fit statistic that was proved useful in discriminating cheaters.

2. METHODOLOGY

2.1 Procedure

The overall rationale of this research is to develop a 'bootstrap' process in which a detector that is trained on one type of cheating is used to build a more general classifier that can detect other types of cheating as well. In our case, the first

[1] https://www.edx.org/edx-terms-service

type is CUMA, and the other type of cheating is unauthorized collaboration, for which we also have a detector that serves as a reference model.

To test this approach, we use the following cross-validation procedure (with K=3, repeated 500 times). First, we train a classifier on a test set, under the assumption that we know to detect only CUMA. In practice, this means that in the test set only CUMA users appear as positive examples (though it may include collaborators data, depending on the random assignment to training/held-out datasets). Second, we use this classifier to classify the held-out dataset. On this set, we check the recall with respect to collaborators. That is, we compute how many of the collaborators in the test set were classified as 'cheaters' by the algorithm, and compare it to the fraction of non-cheaters that were identified as cheaters.

In addition, we evaluate the performance of a classifier that is built on a dataset in which both types of cheaters are tagged as positive examples. The rationale for this evaluation is to support future research, in which we intend to check the generalizability of this classifier not only from one method of cheating to the other, but also between MOOCs.

2.2 Data

We use data from an Introductory Physics MOOC offered by the third author through edX on summer 2014. The course consists of 12 required and 2 optional weekly units. A typical unit contains three sections: Instructional e-text/video pages (with interspersed concept questions, aka checkpoints), Homework, and Quiz. Altogether the course contains 273 e-text pages, 69 videos, and ∼1000 problems. About 13500 students registered to the course, and from them, 502 earned a certificate. This research use the data of 495 certificate earners (7 were omitted due to technical reasons).

2.3 Detecting Cheaters

We define as cheaters those users who use methods that break the code of honor (such as creating multiple accounts or sharing responses with peers) to achieve credit in a way that does not rely on learning. We have algorithms that can detect two specific types of cheating – CUMA, and unauthorized collaboration.

2.3.1 Copying Using Multiple Accounts (CUMA):

To detect CUMA users, we use the algorithm of [2]. It detects 65 users (∼13% of the certificate earners).

2.3.2 Collaborators:

To detect collaborators, we use the algorithm of [16]. Overall, it detects 20 of the certificate earners. However, among those learners, 11 were also classified as CUMA users by the previous algorithm. In these cases, we decided to give priority to the CUMA algorithm, as it represents a more specific behavioral pattern. Hereafter, we refer as 'collaborators' to the 9 accounts who were not CUMA users.

2.4 Feature Engineering

We use the following features, divided into three groups:

Video use:

The rationale for this set of features is that cheaters tend to spend less time on learning resources [1]. As videos are the main learning resource in most MOOCs, this feature can generalize between courses.

i. Watching time: (Log of) The total amount of time, in seconds, that the user spent watching videos.

ii. Fraction of videos watched: The fraction of videos watched. A video is considered as 'watched' if the user played more than 30 seconds of it.

Students Performance:

iii. Correct on first attempt: The fraction of the items that were solved correctly on first attempt.

iv. Mean time to correct: The average time on task, for items solved correctly.

v. Fraction of correct-in-less-than-30 seconds: The fraction of the items that were solved correctly in less than 30 seconds. The 30-second threshold is taken from [14].

Person-fit statistics:

vi. Guttman Error (GE): The number of item pairs in which an easier item is answered incorrectly and a more difficult item is answered correctly, normalized by the total number of pairs [11]. To make our method more general, we use the non-parametric variant, as parametric models (e.g. 2PL IRT) are difficult to fit on MOOCs data. It is computed in R using the package $PerFit$ [19]

vii. Guttman Error on time-on-task (GE-time): This is a new aberrant behavior person-fit statistic that we propose. It basically applies the notion of Guttman Error to time-on-task. It is described in more detail in the Appendix.

2.4.1 Z-scores and Feature Selection

The independent variables were standardized using z-scores, to enable comparing the relative importance of features based on standardized logistic regression coefficients [12], and to allow (in the future) generalizing to other MOOCs. For feature selection, we use a LASSO logistic regression and pick the features that have a non-zero coefficient (the tuning parameter lambda is chosen via cross-validation). This is done in R using the package $glmnet$ [6].

3. RESULTS

This section is organized as follows. First, we report on the results of the feature selection. Second, we present the distribution of the features among CUMA, collaborators, and non-cheaters. Third, we present the performance of the classifier trained on the CUMA users, when used to detect collaborators. Fourth, we report on the performance of a classifier that is trained on both type of cheating.

3.1 Feature Selection

The features with non-zero value are GE, GE-time, fraction of videos watched, and fraction of questions answered in less than 30 seconds.

3.2 Group Differences

Figure 1 presents the differences between the three groups – CUMA users (red), collaborators (black), and non-cheaters (blue), with respect to the four features.

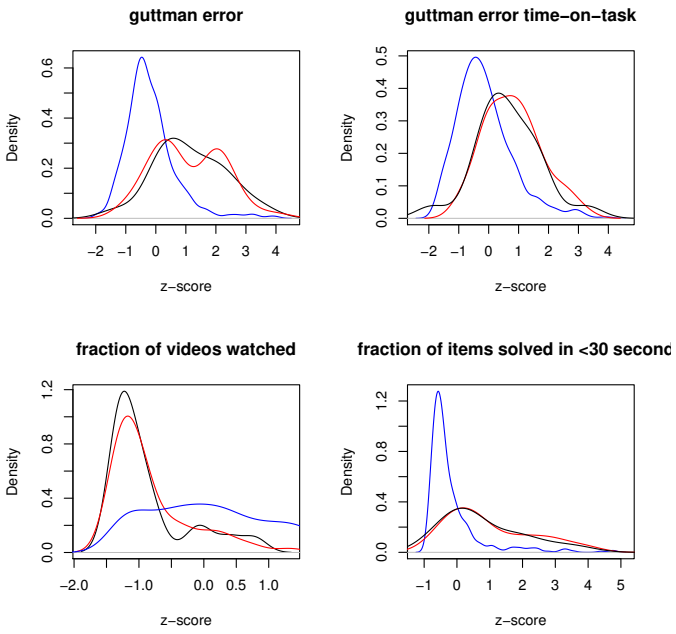

Figure 1: Distribution of the independent variables among CUMA users (red), collaborators (black), and non-cheaters (blue).

3.3 Detecting unauthorized collaboration with a classifier build on CUMA

As described in Section 2, we train a logistic regression model that receives a training set with only 'CUMA' users tagged as positive examples, and use it to detect 'collaborators'. To ensure that we accurately simulate a scenario in which no information on collaborators exists during the training phase, the training set is used for fitting the model and tuning hyper-parameters, and for model-selection. Collaborators might exist in the training data (depending on the random assignment to training/test), but as negative examples (i.e., non-cheaters).

The results ($recall = \frac{TP}{TP+FN}$) of applying this cross-validation process with $K = 3$, repeated over 500 times, are presented in Figure 2. Overall, $mean(recall) = 0.72$, $sd = 0.12$.

For negative examples (neither CUMA nor collaborators), the mean amount of miss-classification ($\frac{FP}{FP+TN}$) is 0.16 ($sd = 0.01$). This means that a collaborator is 4.5 times more likely to be classified as 'positive', than a non-cheater.

3.4 Building a general classifier

Next, we turn to build a classifier on a dataset that contains both types of cheaters as positive examples. The rationale that underlies this is to build a classifier that can 1) 'bootstrap': use data that includes two types of cheating to discover additional ones; and 2) build a global classifier that can (hopefully) generalize across MOOCs.

1. ***Feature Selection.*** This yields the same set of features as reported above.

2. ***Performance of the classifier.*** We evaluate the performance of the classifier using cross-validation and by observing the in-sample classification error. For cross-validation, we measure the AUC of 500 5-fold cross-validation runs. The results are presented in Figure 3. $mean(auc) = 0.85$, $sd = 0.01$.

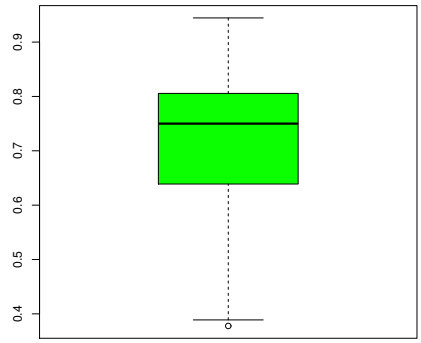

Figure 2: Fraction of collaborators identified by the classifier.

Table 1: Confusion matrix.

		Predicted: False	Predicted: True
Actual	False	377	45
	True	22	51

The confusion matrix for in-sample classification is given in Table 1. The classifier identifies 45 additional users as 'cheaters'. Applying the previous results, we can hypothesize that among these, ~ 35 are 'real' cheaters who are not detected by our previous algorithms, and that ~ 10 are true false positives.

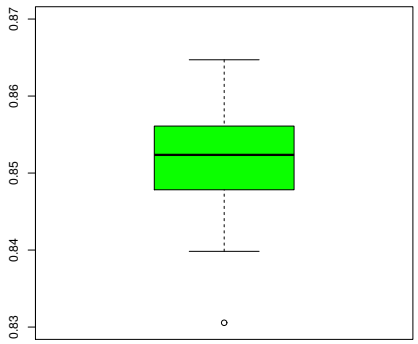

Figure 3: AUC of the classifier

.

4. DISCUSSION

In previous work [1] we *hypothesized* that anomaly detection can be used to build classifiers that can generalize from

known cheating methods to *unknown* ones. However, in our previous work we were unable to provide an empirical evidence for this, due to lack of reference model on a second type of cheating.

The current paper re-visits this approach, exploiting the fact that we now have a dedicated algorithm for detecting unauthorized collaboration, which serves as a reference model. Based on this, we demonstrates that an anomaly-detection based classifier can generalize from one type of cheating to another with high accuracy.

The classifier uses 4 aberrant behavior features. One of them is a new time-based aberrant behavior person-fit statistics that we propose, which was found to be very effective in discriminating cheaters. We name it *Guttman Error-time*.

The power of our approach lies in the fact that 1) it does not rely on prior assumptions on the cheating method, and thus does not require dedicated algorithms that are tailored to a specific method; and 2) the features that it uses are relatively simple to compute, and do not rely on fitting sophisticated parametric models (e.g., IRT). This makes our method scalable and easy to implement across contexts.

Future research. In the future we intend to study whether this method can generalize not only between different methods within the same course, but also between courses.

5. ACKNOWLEDGMENTS

GA's research is supported by the Israeli Ministry of Science and Technology under project no. 713257.

6. REFERENCES

[1] G. Alexandron, S. Lee, Z. Chen, and D. E. Pritchard. Detecting cheaters in moocs using item response theory and learning analytics. In *UMAP'16*, 2016.

[2] G. Alexandron, J. A. Ruipérez-Valiente, Z. Chen, P. J. Muñoz-Merino, and D. E. Pritchard. Copying@Scale: Using harvesting accounts for collecting correct answers in a MOOC. *Computers & Education*, 108:96–114, 2017.

[3] E. W. Black, J. Greaser, and K. Dawson. Academic dishonesty in traditional and online classrooms: Does the. *Journal of asynchronous learning networks*, 12:23–30, 2008.

[4] S. Davis. Academic dishonesty in the 1990s. *The Public Perspective*, 1993.

[5] S. I. De Freitas, J. Morgan, and D. Gibson. Will moocs transform learning and teaching in higher education? engagement and course retention in online learning provision. *British Journal of Educational Technology*, 46(3):455–471, 2015.

[6] J. Friedman, T. Hastie, and R. Tibshirani. Regularization paths for generalized linear models via coordinate descent. *J. of Statistical Software*, 33(1):1–22, 2010.

[7] L. Gibbs. Yes, plagiarism: How sad is that. *Coursera Fantasy: Blogging My Way through a MOOC*, 2012.

[8] C. G. King, R. W. Guyette Jr, and C. Piotrowski. Online exams and cheating: An empirical analysis of business students' views. *J. of Educators Online*, 2009.

[9] D. L. McCabe and L. K. Trevino. Academic dishonesty: Honor codes and other contextual influences. *J. of Higher Education*, 1993.

[10] D. L. McCabe, L. K. Trevino, and K. D. Butterfield. Cheating in academic institutions: A decade of research. *Ethics & Behavior*, 11(3):219–232, 2001.

[11] R. R. Meijer. The Number of Guttman Errors as a Simple and Powerful Person-Fit statistic. *Applied Psychological Measurement*, 18(4), 1994.

[12] S. Menard. Six approaches to calculating standardized logistic regression coefficients. *The American Statistician*, 58(3):218–223, 2004.

[13] C. G. Northcutt, A. D. Ho, and I. L. Chuang. Detecting and preventing "Multiple-Account" cheating in massive open online courses. *Computers & Education*, 100:71–80, 2016.

[14] D. J. Palazzo, Y.-J. Lee, R. Warnakulasooriya, and D. E. Pritchard. Patterns, correlates, and reduction of homework copying. *Phys. Rev. ST Phys. Educ. Res.*, 6, 2010.

[15] J. Reich and J. A. Ruipérez-Valiente. The MOOC pivot. *Science*, 363(6423):130–131, 2019.

[16] J. A. Ruipérez-Valiente, S. Joksimović, V. Kovanović, D. Gašević, P. J. Muñoz-Merino, and C. Delgado Kloos. A data-driven method for the detection of close submitters in online learning environments. In *Proceedings of WWW'17 Companion*, pages 361–368, 2017.

[17] J. A. Ruipérez-Valiente, P. J. Muñoz-Merino, G. Alexandron, and D. E. Pritchard. Using machine learning to detect 'multiple-account' cheating and analyze the influence of student and problem features. *IEEE Transactions on Learning Technologies*, 2017.

[18] D. Shah. By The Numbers: MOOCs in 2018. Class Central, 2019.

[19] J. N. Tendeiro, R. R. Meijer, and A. S. M. Niessen. PerFit: An R package for person-fit analysis in IRT. *Journal of Statistical Software*, 74(5):1–27, 2016.

APPENDIX: Guttman Error-time – a new time-based person-fit statistic

Guttman Error-time applies the idea of *Guttman Error* to time-on-task. Let us define the following notations:
$mean\_ttc(u)$ = the mean time-to-correct of user u on all the items u solved correctly.
$ttc(u,i)$ = time-to-correct of user u on item i.
Now assume we have the Time-To-Correct matrix TTC with $TTC[u,i] = ttc(u,i)$
Let us build a new matrix Boolean-TTC, such that:
Boolean-TTC[u,i] = 0 if $ttc(u,i) > mean\_ttc(u)$, 1 otherwise.
This means that an item on which many students were slower than usual, will have a lot 0's in its column. Intuitively, this is the equivalence of a 'hard' item in the correct-on-first-attempt matrix. Now, we define:
$GE - time = GE_{normed}(Boolean - TTC)$