Ouroboros: Early identification of at-risk students without models based on legacy data

Martin Hlosta[1, 3] Zdenek Zdrahal[1,2] Jaroslav Zendulka [3]

Knowledge Media Institute [1]
The Open University, Walton Hall
Milton Keynes, MK7 6AA, UK
{martin.hlosta;
z.zdrahal}@open.ac.uk

CIIRC, [2]
Czech Technical University
Zikova street 1903/4
Prague, 166 36
Czech Republic

Faculty of Information Technology[3]
Brno University of Technology
Bozetechova 2, Brno, 61266
Czech Republic
{ihlosta; zendulka}@fit.vutbr.cz

ABSTRACT

This paper focuses on the problem of identifying students, who are at risk of failing their course. The presented method proposes a solution in the absence of data from previous courses, which are usually used for training machine learning models. This situation typically occurs in new courses. We present the concept of a "self-learner" that builds the machine learning models from the data generated during the current course. The approach utilises information about already submitted assessments, which introduces the problem of imbalanced data for training and testing the classification models.

There are three main contributions of this paper: (1) the concept of training the models for identifying at-risk students using data from the current course, (2) specifying the problem as a classification task, and (3) tackling the challenge of imbalanced data, which appears both in training and testing data.

The results show the comparison with the traditional approach of learning the models from the legacy course data, validating the proposed concept.

CCS Concepts

•Information systems → Data analytics; •Computing methodologies → *Supervised learning by classification;*

Keywords

Student Retention, Predictive Analytics, Self-Learning, Imbalanced data, Learning Analytics

1. INTRODUCTION

Student dropout is a critical problem which is being tackled by various educational institutions, i.e. universities, high-schools or various platforms for Massive Open Online Courses (MOOCs). According to [21, 25] the number of students not finishing university in Europe is between 20 and 50 %. In USA 20 % of high-school students fail to finish their studies in time [14]. For distance education, these numbers are even more pessimistic with 78% of students not finishing the degree [22]. And even worse, for MOOCs, the percentage of students who registered and successfully completed the course is only 15% on average [12] or even 5% reported by [16]. The problem of identifying students likely to fail the course has been in recent history intensively investigated by the research community [9, 27, 13, 15, 8]. It was also the topic of the KDD'CUP 2015 competition, which mainly focused on predicting students withdrawing from courses in XuetangX, thr Chinese MOOC learning platform[1].

Identifying students, who are at risk of failing or withdrawing from their course, is the first step in the process of providing them with the remedial support. Typically, interventions are mediated by a tutor who receives the results of the predictions. [9, 27]. Alternatively, the prediction system may generate email messages that are sent directly to the student [6]. The primary goal is to improve the students' learning, to retain them in the course, and to help them finish the study programme.

In distance education, most courses are delivered through a Virtual Learning Environment (VLE). In this case, the students' interactions with the VLE are recorded and stored. Besides, student data include demographic information, assessment results, etc. Together, these sources provide a large amount of student data for analysis. After cleaning and pre-processing the data, machine learning techniques are commonly used to build predictive models. These are then utilised to provide the predictions of at-risk students.

A typical approach is to train the models using legacy data from a previous presentation of the course[27]; the models are then applied to the current presentation. However, this approach cannot be used for new courses which have no history. In such a case, it is necessary to find a different solution.

The highest level of dropout typically happens in the first years' courses, and many students drop out even during the first few weeks of the course presentation. This finding has been confirmed by the analysis of both distance Higher Education (HE) courses [27] and MOOCs [23]. One of the explanation is that the drop-out might happen due to the possible fee reimbursement. The withdrawal rate is shown in

LAK '17, March 13 - 17, 2017, Vancouver, BC, Canada

ISBN 978-1-4503-4870-6/17/03…$15.00

DOI: http://dx.doi.org/10.1145/3027385.3027449

[1]KDD CUP 2015 – http://kddcup2015.com

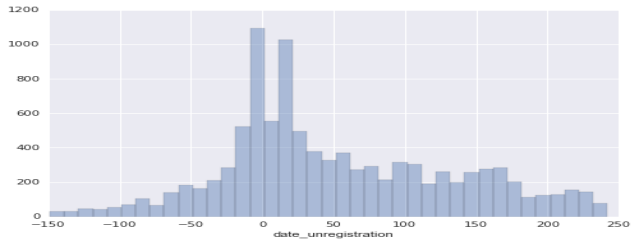

Figure 1: Number of withdrawn students for 7 courses in days relative to the start of the course (day=0).

Figure 1 for the HE OULAD dataset[2]. Therefore, the goal is to identify at-risk students as early as possible. It's worth noting that the same pattern might not necessarily hold for all the educational institutions - based on the course design, significant student dropout may sometimes happen later in the course [15].

2. IDENTIFYING AT-RISK STUDENTS: RELATED WORK

The results of the state-of-the-art solutions are highly determined by the data available for analysis, which is dependent on the type of the educational institution (i.e. whether it is a high school, university, distance learning university or MOOCs). Nevertheless, the main idea is usually the same, i.e. to use legacy data to train predictive models. The features can be selected either by tutors experience or by machine learning algorithms. Then, the models are used to provide predictions for the current students - current presentation of the course, current cohort of high-school, etc.

2.1 High schools

High schools in the USA are usually interested in forecasting whether students will finish their studies in time. Using the data from the previous cohort, models are trained and applied to the current cohort. The precision of the models increases when students approach the final grade and thus the prediction time frame decreases [18, 11]. In a recent study GPA[3] was found to be the feature with the highest predictive power [18].

2.2 Higher Education

In [8, 19, 13], students' previous results were used to train the models for identifying success or failure of the current cohort. In contrast, [26] extracted the predictive features from demographic data only. Most of the recent work in Higher Education makes use of demographic, performance and behavioural data extracted from the VLE. The key issue addressed by different educational institutions and researchers is how the concept of an at-risk student is defined. It can be a student with the grade lower than C [9] or even B- [2], less than 60% [3], not submitting the following assessment [27], or dropping out in the following days [15, 23].

2.3 MOOCs

In MOOCs, the main source of information is usually the click-stream data, as the current performance and demographic may not be available. The key point is to tackle the high dimensionality of the data. Students daily interact with a lot of study material. The activity types include viewing videos, reading study texts, posting in forums etc. The predictions are usually based on the summary of clicks, possibly grouped by activity types [28]. MOOCs differ from HE courses: students are often not motivated to finish the MOOC course and they may register only to have access to videos or text materials, typically provided for free [28].

The current research also differs in defining the performance measure used to evaluate the methods. Frequently used are ROCAUC[4] [18, 1, 7], Precision and Recall [27, 9, 2], less often Accuracy [10], but mostly it's a combination of AUC and Precision/Recall.

Sometimes, various constraints were posed to the problem, e.g. focusing on obtaining smoothed probabilities across the predicted weeks [7] or limiting the predictions to most at-risk students. Top-K-Precision and Top-K-Recall were used in [18] where K defines the percentage of students selected as at-risk ordered by probability of failing and used for calculating Precision and Recall. This study discussed the possibility of limited resources to assist students. In this case, the schools were able to provide support to at least 5% of the student population. Due to different problem specifications, data used, and evaluation metrics, a comparison of existing solutions is not an easy task.

2.4 Early identification of at-risk students

When all data are available, the best predictor makes use of actual performance either by: (a) student study history measured (e.g. by GPA) or (b) by evaluating the progress in the current course from assessment results. However, student study history - is not available for entry-level courses, or more generally courses, which are taken at the beginning of the study programme. Moreover, these courses require increased attention because student dropout is typically high.

The progress in the current course is unavailable before the first assessment (denoted as A1) is evaluated, though A1 is often important for early predictions. This issue has been addressed in [27] by predicting submissions of the A1 from demographic and pre-A1 VLE activities with models trained on the previous presentation of the same course. To early identify at-risk students, [10] used behaviour in the first week, by evaluating quiz results as the most important attribute. Similarly, [28] has found quizzes to be most informative for the predictions.

3. PROBLEM SPECIFICATION

3.1 Importance of the first assessment

It is important to investigate whether A1 is a good predictor of the overall success in the course. We assume that the student succeeds in A1 if he/she submits A1 and achieves a score higher than 50% of the points. This has been investigated for courses A to G, see table 1. The results are divided into 4 columns: (a) probability of failing the course given student failing A1 (scoring less than 50 %), (b) probability of failing in the course given that the student did not submit A1, (c) number of students who failed the course and

[2]https://archive.ics.uci.edu/ml/datasets/Open+University+Learning+Analytics+dataset
[3]GPA = Grade Point Average
[4]Receiver Operation Characteristic Area Under Curve

submitted but failed A1, and (d) number of students failing the course and not submitting A1. The numbers in the table are means calculated across all presentations available in the OULAD dataset.

Based on the previous presentation, it's possible to extrapolate the probability of failing based on the A1 in the next presentation[5]. If no presentation is given we can extrapolate from all courses.

The probability of failing the course if a student hasn't submitted A1 is almost 90%, making A1 a strong predictor of future failure. If no data from the previous presentation of the course is available, it's impossible to use the results from the assessment before they are marked. However, even the assessment submission is a good predictor for identifying at-risk students. On average, there is 95% probability that a student will not finish the course given that he/she hasn't submitted A1. When limiting the approach only to predict submissions instead of predicting the failures, we are not able to identify some of the at-risk students. According to the 3rd and 4th column in Table 1, we are missing 1392 students (i.e. 15% of those that fail), but at the same time, the probability of failing the course is 5% higher, making the predictions more accurate.

Table 1: Probability of failing the course (F) if failed (F_A1) or not submitted (NS_A1) A1 for courses averaged for all available presentations and count of students.

Course	$P(F\|F\_A1)$	$P(F\|NS\_A1)$	$CNT(F \wedge F\_A1 \wedge S)$	$CNT(F \wedge NS\_A1)$
A	0.8004	0.9421	23	50
B	0.8809	0.9905	434	1888
C	0.8381	0.8252	255	1710
D	0.9651	0.9876	431	1436
E	0.9808	0.9932	49	643
F	0.9805	0.9930	122	1527
G	0.8300	0.9157	78	453
AVG / SUM	0.8966	0.9496	1392	7707

3.2 Assessment description

Each assessment has a cut-off date before which the students have to submit their assignment. Four types of data are available:

1. students' demographic information (age, gender, etc.),
2. students' interactions with the VLE system,
3. information about students' date of registration and
4. a flag indicating student assessment submission.

The latest available data always come from the previous day, no information is available for the current day, i.e. we know students' activities and whether she/he has submitted by the end of the previous day.

[5]The values for each presentation has been omitted for space but they are similar across presentations.

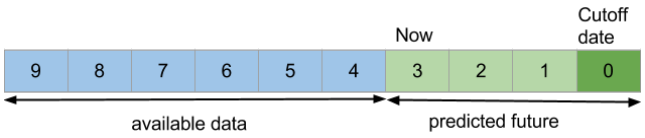

Figure 2: Time line with the current day and cut-off date.

3.3 Dealing with lack of legacy data

There are two straightforward possibilities how to deal with the lack of data from previous presentations.

It is possible to build a prediction model based on all available courses. However, it has been shown in [27] that the identification of at-risk students is more accurate when the predictions are tailored for each course separately.

Another option is to use data from the students' previous study results. Unfortunately, these data are not available for the courses at the beginning of the study programme. At the same time, these level-1 courses usually have lower retention [27], therefore they are more important to consider them for analysis.

3.4 Ouroboros: Self-Learning approach

This paper proposes a new *Self-Learning* approach, i.e. to use only data from the running presentation for training predictive models. The underlying idea is to use the data about students who have already submitted the next assignment and exploit the patterns of their behaviour to identify the students who might be at risk of not submitting. It's expected that the behaviour of learners who are about to submit will follow a similar pattern as those who have already submitted and differs from students who will not submit.

There are several options how to make use of these patterns. In this paper, we define the task as a binary classification problem: Given the current day, which is n days before the cut-off date, the goal is to construct a binary classification model that will predict whether the student (1) will submit or (2) will not submit the next assessment in time, i.e. today or within the next n days. If $n = 0$, predictions are made on the cut-off day. Only students that are registered in the course and haven't submitted the assessment yet are subject to the prediction. The figure 2 depicts the problem for $n = 3$.

4. OUROBOROS FRAMEWORK

Let's denote the cut-off date as *cutoff_date* and the date when the prediction is made, which is n days before the cut-off day, as *prediction_date*. In order to be able to create the prediction model for interval [*prediction_date*; *cutoff_date*] we need labelled examples for interval of the same size [*d_prediction_date*; *d_cutoff_day*] such that *d_cutoff_date* = *prediction_date*. The *d_prediction_date* and *d_cutoff_day* can be considered to be a dummy prediction and cut-off day, respectively.

The example of the problem is depicted in Figure 3 in the top part a). Here, the cut-off date is within 3 days from the current day and we want to predict if students submit either today or within the next 3 days. The data for the current day are unavailable, so the training data will come from the

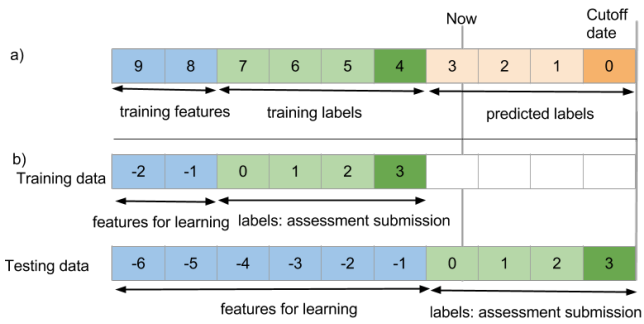

Figure 3: Classification framework for self-learning and testing predictions of at-risk students

days $[presentation\_start; now + 5] = 8$ with the labels of submission in $[now + 4; now + 1] = [7; 4]$.

The bottom part of Figure 3 b) shows the relative view of the days for training and testing data, $day = 0$ denotes the current day, negative indexes relate to known data and positive indexes to unknown. Thanks to this view it's visible that though we have more days available when applying the predictive model, some older days cannot be used since they were not present in the training phase.

4.1 Extending labelling window

Based on the described concept, going back in history means the window for labels is growing. The more days before the cut-off date, the more days for training labels we need. The situation for the current day being 0 to 3 days before the cut-off date is depicted in the Figure 4. For n days before the cut-off, the size of the window both for training and testing labels is $n + 1$.

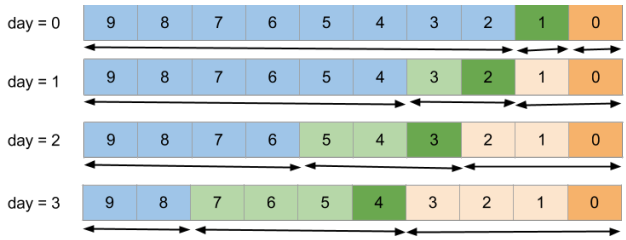

Figure 4: Extending window for training and testing labelled data. Day = n denotes that the current day is n days from the cut-off date, (day=0 cut-off day is today, day=1 cut-off day being tomorrow, etc.)

4.2 Features for learning

The available data for learning include information about student demographics and activities in the VLE. As the demographic data is static, it is only necessary to perform transformations, such as vectorisation of categorical data and standardisation/normalisation for numerical data.

On the other hand, the VLE data are very rich containing daily click summary activities grouped by specific activity, for example "student A viewed 10 times the specific PDF resource *study_material.pdf*". All the activities are grouped into *activity types*, so all the PDF resources are grouped as a resource. There are approximately 30 different activity types such as *forum, video, resource*, etc.

Given the current day when the model is learned, the VLE features are aligned backwards in time on this day, i.e. *day_0* is the current day, *day_1* is referring to yesterday etc. The oldest day used for training is the day that the course starts.

In addition to VLE daily counts, it's possible to extract various summarising statistics about student behaviour in the VLE, such as the number days that the student was active in the VLE (i.e. when he/she at least logged in). These statistics and all the features are described in Table 2.

5. PREDICTIVE MODELLING

For training the models and for the whole evaluation framework, the Python Scikit-learn library [20] was used, which provides a large number of existing implementations of classification algorithms and preprocessing routines.

For training the models, we chose models that support probabilistic predictions. This enables us to order students according to their likeliness to fail, and then apply the resources limitation. Also, the existing results from the research in the identification of at-risk students were taken into consideration. The selected algorithms included: Logistic Regression (LR), Support Vector Machines (SVM), Random Forest (RF), Naive Bayes (NB), and the Tree Boosting XGBoost. The last one was selected due to its success in many Kaggle[7] competitions. According to [4] 17 out of 29 winning solutions in 2015 used XGBoost. Moreover in the KDD-CUP15 focused on predicting students' dropout, all top 10 solutions used this algorithm [8].

6. TACKLING IMBALANCED DATA

Machine learning algorithms are usually designed to learn concepts from data when the classes in the training data are balanced. However, in many real-world problems, the dataset includes a class with a significantly lower number of instances than the others. Without any changes, these algorithms perform poorly and therefore new approaches have been developed [6].

The basic approaches to deal with imbalanced data address the problem at the following two levels:

- *Data level* – using various sampling methods to modify the class distribution in a way that the training data are balanced.
- *Algorithmic level* – cost-sensitive learning, One-class classification methods, and various ensemble methods are among those mostly used.

  The key idea of *cost-sensitive learning* is to penalise the cost of error on the minority class, which is incurred during the training phase. This can be achieved by specifying a *cost matrix*. However, for the binary classification problem, it's usually good enough to set the weight for the minority class (with the assumption that the weight for the majority class remains 1).

The problem of imbalanced data appears in the existing research in predicting at-risk students. In [9] a data

[7]https://www.kaggle.com
[8]https://www.linkedin.com/pulse/present-future-kdd-cup-competition-outsiders-ron-bekkerman

Table 2: Features used for learning the model.

No.	Type	Dim.	Description	Examples
1	Demographic	8	Static demographic data	Age, IMD[6], Qualification, Region, Gender, Declared Disability, Number of previous attempts, number of currently studied credits
2	Registration info	1	The registration day relative to start of the course - positive or negative number	-
3	VLE statistics	28	Various statistic measures about student behaviour in VLE	1) Num. of consecutive days that the student is currently active, 2) first and last day he/she was active or indication of never logged in, 3) average/median of clicks and number of materials visited per day normalised either by all days or only days when he/she was active in the VLE, 4) total number of active days in the VLE
4	VLE statistics before presentation start	19	Same as 3), measured before only the start of the presentation	Same type of statistics as for previous feature type (4) but only limited to 3) and 4) features mentioned in the examples.
6	VLE daily counts per activity type	50-560	Number of clicks in the VLE grouped by activity type per day	Number of clicks in resources/forum in day 0, 1, ...

level approach was used, combining random over-sampling and under-sampling. Similarly, [24] improved the AUC and F1-Score by over-sampling the dataset using SMOTE algorithm. Moreover, they examined which algorithm best copes with different cost ratios specified to False Negatives and False Positive errors. On the other hand, in [7] the problem was tackled by focusing only on 'active' students and completely omitting those who haven't shown interest in doing assignments.

6.1 Problem and solution

The specificity of this problem comes from the fact that the ratio between the majority and the minority class is changing in time. The more we move backwards from the cut-off date, the higher is the imbalance ratio in the training data, because there are fewer students who have already submitted the assessment and also more students that withdraw later in the course meaning they don't appear in the training data. Most important, the majority class in the training data is minority class in the testing data. The ratio between the classes is depicted in the Figure 5.

For the algorithms such as SVM and Logistic Regression, it's possible to use cost-sensitive learning by specifying the weights of the classes during the training. The most suitable way proved to be to set the weights proportionally to the ratio of the cardinality of minority and majority classes. Moreover, several ensemble based algorithms, which are able to cope with the imbalance data. for learning were used.

The important question, when dealing with imbalanced data is the selection of the performance metric for algorithms comparison. The area under ROC curve ($ROCAUC$) and area under the Precision-Recall curve ($PRAUC$) are the most suitable measures. The latter is giving more information about the algorithm performance on the target class, especially when the data are imbalanced and the target class is more important [5]. Moreover, this metric suits more the problem of identification at-risk students when Precision is more important metric to measure than FPR, which is used in ROC. For these reasons, we chose $PRROC$ as our evaluation metric.

Figure 5: Ratio for NotSubmit class for training and testing data.

7. EXPERIMENTAL RESULTS

The proposed framework for learning the student dropout model has been evaluated using various experiments with all the data and code publicly available.

7.1 Experimental setup

The experiments were conducted on four level-1 university courses with 1200 to 2500 students on the publicly available OULAD - Open University Learning Analytics Dataset [17].

For all the courses, the goal was to predict the submission for A1, with the cut-off ranging from day 19 to 33. More information about the courses is available in the Table 3. We narrowed the focus on the most recent 2014 presentation, but the numbers don't differ much.

The courses cover wide range fields such as maths, engineering, history or social care. They last between 20 and 30 weeks and they are organised in logical blocks, each of them completed by an assessment. In order succeed in a course, students have to achieve minimum scores in the assessments and then pass the final exam.

Three more courses are available in the dataset, A, C and G, A being level-3 course and G being a preparatory course. These courses have different properties and we omitted them from the comparison. Course A and C have a lower number of enrolled students, and the cut-off date of A1 for course G is late in the course, on day 61. Surprisingly, despite only 8% of students submitted A1 in the level-3 course A, the course has the lowest retention out of all courses. The course C was withdrawn from the experiments because this course doesn't have the previous presentation 2013J and we wanted these experiments to be comparable.

Table 3: Information about the courses under analysis - 2014 presentation

Course	Num. of students	Pass Rate [%]	A1 S/NS[%]	Class Ratio for cut-off	Cut-off date
A	365	30.69	92.23	12.04	19
B	2292	49.74	77.31	3.41	19
C	2498	59.37	57.04	1.33	32
D	1803	56.07	78.48	3.65	20
E	1188	42.42	78.20	3.59	33
F	2365	52.77	77.12	3.37	24
G	749	40.72	77.97	3.54	61

The experiments were focused on the following goals:

1. Daily analysis of classification performance across days and machine learning algorithms.
2. Tackling the problem of imbalanced data.
3. Compare Ouroboros against models trained on legacy data, i.e. previous presentation of the same course.
4. Analysis of Precision-at-K for various K to see these metrics for a limited resources for interventions.
5. Feature importance for the best algorithm from the first experiment to see the change in time and across courses.

The source code of the Ouroboros framework together with all the performed experiments and scripts for the presented statistics are available on GitHub[9].

7.2 Daily results

This experiment focused on comparing the performance of the algorithms with each other for various days relative to the cut-off. The main goal was to observe a change of performance when moving further back to the history and which machine learning models are coping best with the given data.

The Table 4 shows the $PRAUC$ for the used classification methods. The value for the classifier is an average computed over the four courses. The table shows the performance in the cut-off date and up to 11 days before the cut-off date.

The performance is better for the cut-off date and then drops down when going back in time, especially in the day 1 and 2. The highest $PRAUC$ was achieved by three models - XGB, RF and weighted SVM with RBF kernel. While SVM performed best in the cut-off date, RF in day 1 and 2, XGB gave better results from day 3 to 11.

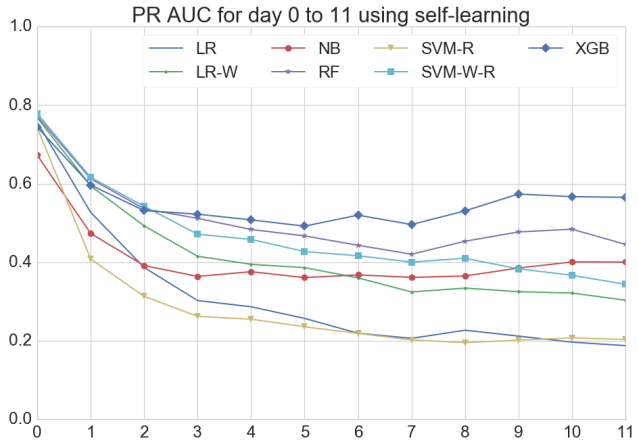

Figure 6: $PRAUC$ for days 0 to 11 before the cut-off. LR stands for Logistic Regression, with W indicating weighted LR used for imbalanced data, same for SVM. SVM-R and SVM-W-R = SVM with RBF kernel, RF = Random Forest, XGB=XGBoost, NB=NaiveBayes

7.2.1 *Dealing with imbalanced data*

Both the Figure 6 and the Table 4 reveals how important might be setting class weights for the machine learning algorithm to handle the imbalanced data. The difference in performance between weighted and unweighted versions of LR and SVM becomes visible when moving further from the cut-off date and having higher imbalance ratio. The performance of the weighted version doesn't suffer from the change that much because the error made on the minority class influences the model. Given highly imbalanced data, the model might not be able to underpin minority class and classify all the data to the majority class.

Moreover, we utilised several sampling methods for modifying the class distribution of the data from the ImbalancedLearn[10] but even using sophisticated sampling methods didn't lead to better results than for the class weighting and using ensemble methods.

7.3 Comparing with learning from legacy data

The aim of this experiment was the comparison of the self-learning approach with training on the legacy data. In the real world, there might not be any previous course to

[9]https://github.com/hlostam/ouroboros_paper/

[10]ImbalancedLearn – github.com/scikit-learn-contrib/imbalanced-learn

Table 4: ***PRAUC*** **values for different days trained on the same presentation.**

Day	SVM-W-R	SVM-R	LR	LR-W	NB	RF	XGB
0	**0.7790**	0.7435	0.7561	0.7682	0.6779	0.7748	0.7442
1	0.6161	0.4081	0.5267	0.5944	0.4587	**0.6184**	0.5965
2	0.5436	0.3138	0.3852	0.4934	0.3673	**0.5353**	0.5315
3	0.4726	0.2629	0.3019	0.4164	0.3412	0.4960	**0.5225**
4	0.4596	0.2547	0.2866	0.3954	0.3577	0.4796	**0.5079**
5	0.4289	0.2363	0.2569	0.3870	0.3453	0.4600	**0.4920**
6	0.4171	0.2185	0.2195	0.3610	0.3475	0.4234	**0.5200**
7	0.4024	0.2027	0.2072	0.3263	0.3456	0.4309	**0.4959**
8	0.4118	0.1948	0.2272	0.3350	0.3487	0.4378	**0.5309**
9	0.3850	0.2031	0.2120	0.3260	0.3809	0.4820	**0.5737**
10	0.3677	0.2074	0.1967	0.3225	0.4011	0.4785	**0.5669**
11	0.3440	0.2033	0.1879	0.3039	0.3985	0.4569	**0.5652**

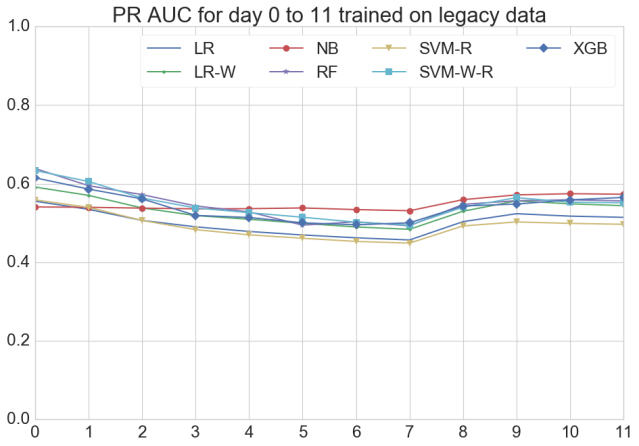

Figure 7: ***PR AUC*** **for days 0 to 11 before the cut-off using training on the previous presentation.**

compare with, but the OULAD dataset has them. The experiments were run for the same courses and same days as in the previous experiment.

Looking at the previous Figure 6 and the Figure 7, the results show that more data provided to the algorithms using training on the previous presentation helped the algorithms to have more stable results. However, when predicting in the cut-off date, the performance of Ouroboros based models was significantly better, around 10%.

7.3.1 Comparing with existing solution

Moreover, we were able to compare our solution with the existing work for predicting next assessment submission in [27], denoted as PREV_MODEL. F1-Score, Precision and Recall was computed in the selected days before the cut-off date because these predictions were computed weekly not daily. F1-score was selected as an evaluation metric as it represents a harmonic mean between Precision and Recall. Both self-learning and learning using legacy data were compared and again using the 4 courses as previously.

Based on the previous experiment, we selected the best algorithm, which was XGBoost and optimised the probability threshold of the predictions on the training data in order to maximise the F1-score. This threshold was used to compute the evaluation metrics.

Table 5 shows that both Ouroboros and our solution trained on the previous presentation outperform the PREV_MODEL in F1-score. Ouroboros performs better in the day 0 while training on legacy data in the other days. The only situation when the PREV_MODEL performed better is course E on the cut-off date.

7.4 Top-K-Precision

Although PR-AUC is a suitable measure for comparing classifiers' performance, the target users are sometimes interested how confident are the classifiers for the top ranked students in terms of their probability to Not Submit the A1. As mentioned, this might be useful for determining the quality of predictions given a limited resources for interventions.

For this experiment, the results are compared with two baseline models. Base[Nonactive] model classifies all the students that haven't accessed the VLE so far as NotSubmit and all the others as Submit. The Base[NotSubmit] assigns all the students to NotSubmit class, meaning that we would intervene with all the students. Those were not used in the previous experiment as they don't provide probabilistic prediction and their performance was otherwise overly optimistic.

The Figure 8 contain 3 sub-figures of precision for the first 5,10 and 25%, and similarly Figure 9 contain Top-K-recall. It's clearly visible that as the k increases the precision decreases, especially when moving from top 10% to top 25%. Again, as the daily gap towards the cut-off date increases the performance goes down. The drop is greatest from day 0 to day 2, thanks to very high precision achieved in day 0. The decrease continues only until the day 7 and 8. We can observe a drop of precision from day 8 to 7 and in some of the models a peak from 9 to 8. This drop can be explained looking at the performance of the baseline classifier, because these days are typical for students with completely no activity so far to withdraw from the course, meaning that this low-hanging fruit disappears from the data and classifiers focused on them drop in performance.

7.5 Intervention strategy recommendation

The predictions are being used by tutors to spot at-risk students and make an appropriate intervention if necessary. Based on the results, we also suggest when might be the right time to intervene.

Given the graphs from 8, we tried to find the most suitable k and day for predictions and interventions with students. Because the drop from top 10 to top 25 %, k=10 seems like a reasonable choice. Intervening in day 0 might be very

Table 5: Ouroboros vs training on the legacy data vs PREV_MODEL using F1-score, Precision and Recall.

Course	Days to cut-off	PREV_MODEL			Ouroboros best			Prev. presentation		
		F1	Prec.	Recall	F1	Prec.	Recall	F1	Prec.	Recall
B	5	0.1741	0.5124	0.1049	0.2808	0.1635	0.9949	**0.3592**	0.3034	0.4400
B	0	0.1633	0.7031	0.0924	**0.6724**	0.5751	0.8093	0.4503	0.3725	0.5692
D	6	0.3072	0.3615	0.2670	0.2596	0.1495	0.9847	**0.3109**	0.1843	0.9924
D	0	0.3740	0.5476	0.2840	**0.5534**	0.3986	0.9048	0.3026	0.1784	0.9960
E	5	0.5678	0.6505	0.5038	0.3511	0.2139	0.9792	**0.5792**	0.6752	0.5072
E	0	**0.6857**	0.7579	0.6261	0.6528	0.5044	0.9247	0.6718	0.6804	0.6633
F	3	0.3931	0.4191	0.3701	0.3366	0.1858	0.9898	**0.5618**	0.5303	0.1711
F	0	0.5134	0.5583	0.4752	**0.7131**	0.6698	0.7624	0.5979	0.6170	0.5800

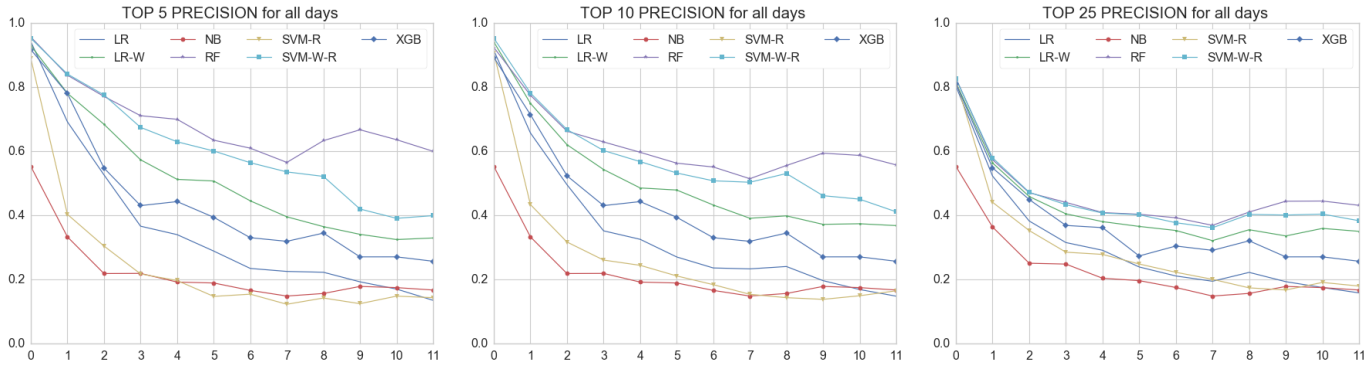

Figure 8: Top-K-Precision for k = 5,10,25

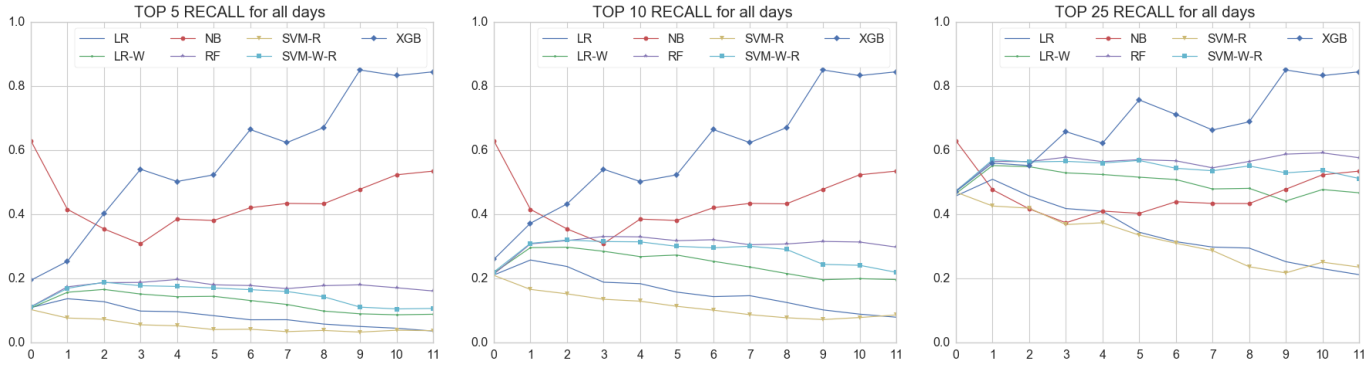

Figure 9: Top-K-Recall for k = 5,10,25

accurate but intuitively it's too late to provide students with any help. There is a drop from day 3 to 4 and from day 6 to 7 in precision but peak for 3 a 6 in Recall for XGBoost.

Given this information, one reasonable strategy might be to use the XGBoost model for Top-5-Precision 6 days before the cut-off and Top-10-Precision model on day 3. Using Ouroboros approach both should provide average precision around 0.4 and Recall 0.6.

7.6 Feature importance

Apart from SVM, most of the used models enable to extract importance of the features used for prediction easily. We selected XGBoost as the best performing classification model on average, especially further from the cut-off. Then, we extracted top 5 ranked features for the analysed courses in days 0, 3 and 7.

Table 6 shows that across all the courses the most important factors are coming from the specific usage of the VLE and the VLE statistics. While VLE statistics prevail across all the selected days, the specific VLE activity type importance varies. On the cut-off date, login information appears among the most important factors and in two of the courses visiting forum becomes important predictor.

8. DISCUSSION

Though the analysed courses come from different fields, the relatively small number of courses under analysis do not allow us to investigate the dependence of the performance on the discipline. For example, Table 5 shows that the lowest

Table 6: Most important features for XGBoost

#	Course	Day0	Day3	Day7
1	B	first.login	days.fromvleopen	days.fromvleopen
2	B	clicks.fromstart	is.click.7.subpage	resources.fromvleopen
3	B	max.mat.beforestart	is.click.6.oucontent	clicks.9.oucontent
4	B	clicks.fromvleopen	clicks.2.forum	clicks.9.subpage
5	B	last.login.rel	avg.mat.cnt.fromstart.peractive	clicks.13.oucontent
1	D	days.fromstart	clicks.fromvleopen	clicks.30.oucontent
2	D	last.login.rel	clicks.1.oucontent	clicks.5.glossary
3	D	studied.credits	resources.fromvleopen	resources.fromvleopen
4	D	clicks.6.forum	clicks.38.oucontent	clicks.4.glossary
5	D	clicks.9.oucontent	clicks.7.glossary	clicks.2.glossary
1	E	last.login.rel	clicks.34.oucontent	clicks.36
2	E	min.click.fromstart.peractive	min.click.fromstart.peractive	clicks.20.quiz
3	E	clicks.3	clicks.11.quiz	clicks.31
4	E	median.mat.cnt.beforestart.peractive	clicks.18.quiz	clicks.6.url
5	E	clicks.fromvleopen	clicks.12.subpage	min.click.beforestart.peractive
1	F	last.login.rel	clicks.fromvleopen	days.fromvleopen
2	F	days.fromstart	min.click.fromstart.peractive	clicks.8.htmlactivity
3	F	clicks.fromvleopen	clicks.33.subpage	is.click.1.oucontent
4	F	days.fromvleopen	clicks.4	is.click.22.ouwiki
5	F	clicks.6.forum	clicks.10.resource	is.click.9.forum

F1 score is for course D, however, a deeper analysis would be required to support the claim that the field of the course D influenced the classifier performance.

8.1 Usage in different contexts

The proposed method is not limited only to A1 at OU and given several conditions it can be used without adaptation for further assessments and in other contexts, such as other distant educations or MOOCs. There need to be (1) a task/event with specified cut-off date and also (2) students that fulfil it in advance. It can also be used to different kind of tasks such as whether students will register for a course. Although we didn't examine this yet, we expect that this approach can be used for a wide class of problems outside the Learning Analytics field, given that they satisfy the conditions mentioned above.

8.1.1 Limitations

When there is no deadline specified, the method would need to be adapted to treat the window for training the model differently. Then we would be able to use the same approach for predicting dropout of students or potentially if students will register for paid certificate in MOOCs. Similarly, for the second condition, the approach wouldn't be suitable for High Schools scenario predicting whether the students will finish the studies in time because the students are not expected to complete it in advance.

9. CONCLUSIONS AND FUTURE WORK

This paper introduced *Ouroboros*, the novel approach to early identification of at-risk students in the courses without legacy data, i.e. data from previous presentations. Our method utilises the importance of the first assessment being a critical milestone in the progress of the course. The key idea is that the learning patterns can be extracted from the behaviour of students who have already submitted their assessment earlier.

We defined the problem as a binary classification task with the goal being able to learn and predict daily using the widening window. The approach was evaluated on the publicly available OULAD dataset using 4 level one courses. The experiments showed that the method can successfully predict at-risk students, for the day 0 and 1 it gives better results than training using the legacy data.

Analysis of feature importance of XGBoost as the best performing algorithm showed that specific VLE activities are important for predicting at-risk students together with statistical information about VLE usage.

In the further work, we want to explore how the self-learning model is performing in the later phases of the courses. Also, we want to combine both models to improve predictions even for the training using the legacy data. The large-scale analysis might reveal the influence of the learning objective or field on the classifier.

Acknowledgement

This work was partially supported by the institutional resources for research by the Czech Technical University in Prague, Czech Republic and The Ministry of Education, Youth and Sports from the National Programme of Sustainability (NPU II) project "IT4Innovations excellence in science - LQ1602".

10. REFERENCES

[1] E. Aguiar, H. Lakkaraju, N. Bhanpuri, D. Miller, B. Yuhas, and K. L. Addison. Who, when, and why: A machine learning approach to prioritizing students at risk of not graduating high school on time. In *LAK '15*, 93–102, New York, NY, USA, 2015. ACM.

[2] J. Bainbridge, J. Melitski, A. Zahradnik, E. Lauría, S. M. Jayaprakash, and J. Baron. Using Learning Analytics to Predict At-Risk Students in Online Graduate Public Affairs and Administration Education. *The JPAE Messenger*, 21(2):247–262, 2015.

[3] R. S. Baker, D. Lindrum, M. J. Lindrum, and D. Perkowski. Analyzing early at-risk factors in higher education e-learning courses. In *Proceedings of the 8th International Conference on Educational Data Mining, EDM 2015, Madrid, Spain, June 26-29, 2015*, 150–155, 2015.

[4] T. Chen and C. Guestrin. Xgboost: A scalable tree boosting system. *CoRR*, abs/1603.02754, 2016.

[5] J. Davis and M. Goadrich. The relationship between precision-recall and roc curves. In *Proceedings of the 23rd international conference on Machine learning*, 233–240. ACM, 2006.

[6] H. He and E. A. Garcia. Learning from imbalanced data. *IEEE Trans. on Knowl. and Data Eng.*, 21(9):1263–1284, Sep 2009.

[7] J. He, J. Bailey, B. I. Rubinstein, and R. Zhang. Identifying at-risk students in massive open online courses. In *AAAI*, 1749–1755, 2015.

[8] S. Huang and N. Fang. Predicting student academic performance in an engineering dynamics course: A comparison of four types of predictive mathematical models. *Comput. Educ.*, 61:133–145, Feb 2013.

[9] S. M. Jayaprakash, E. W. Moody, E. J. M. Lauria, J. R. Regan, and J. D. Baron. Early Alert of Academically At-Risk Students: An Open Source Analytics Initiative. *Journal of Learning Analytics*, 1(1):6–47, 2014.

[10] S. Jiang, M. Warschauer, A. E. Williams, D. ODowd, and K. Schenke. Predicting mooc performance with week 1 behavior. In *EDM14*, 273–275, 2014.

[11] R. A. Johnson, R. Gong, S. Greatorex-Voith, A. Anand, and A. Fritzler. A data-driven framework for identifying high school students at risk of not graduating on time. *Bloomberg Data for Good Exchange Conf.*, 5, 2015.

[12] K. Jordan. Mooc completion rates: The data. http://www.katyjordan.com/MOOCproject.html, 2015. Accessed: 2016-06-10.

[13] R. R. Kabra and R. S. Bichkar. Performance prediction of engineering students using decision trees. *International Journal of Computern Applications*, 36(11):8–12, December 2011.

[14] G. Kena, J. W. X. R. A. Musu-Gillette, Laurenand Robinson, J. Zhang, S. Wilkinson-Flicker, A. Barmer, and E. D. V. Velez. The condition of education 2015. Technical Report 2015-144, NCES, May 2015.

[15] M. Kloft, F. Stiehler, Z. Zheng, and N. Pinkwart. Predicting mooc dropout over weeks using machine learning methods. In *Proceedings of the EMNLP 2014 Workshop on Analysis of Large Scale Social Interaction in MOOCs*, 60–65, 2014.

[16] D. Koller, A. Ng, C. Do, and Z. Chen. Retention and intention in massive open online courses: In depth. EDUCAUSE, http://www.educause.edu/ero/article/retention-and-intention-massive-open-online-courses-depth-0, Jun 2013. [Online; posted 3-June-2013].

[17] J. Kuzilek, M. Hlosta, and Z. Zdrahal. Open university learning analytics dataset. In *Data literacy for Learning Analytics workshop at LAK16, 26th April 2016, Edinburgh, UK*, 9, 2016.

[18] H. Lakkaraju, E. Aguiar, C. Shan, D. Miller, N. Bhanpuri, R. Ghani, and K. L. Addison. A machine learning framework to identify students at risk of adverse academic outcomes. 1909–1918, 2015.

[19] M. Pandey and V. K. Sharma. A decision tree algorithm pertaining to the student performance analysis and prediction. *International Journal of Computern Applications*, 61(13):1–5, January 2013.

[20] F. Pedregosa, G. Varoquaux, and e. Gramfort. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[21] J. Quinn. Drop-out and completion in higher education in europe among students from under-represented groups. Technical report, European Commission, Oct 2013.

[22] O. Simpson. 22% - can we do better? In *The CWP Retention Literature Review*, 47, 2010.

[23] C. Taylor, K. Veeramachaneni, and U. O'Reilly. Likely to stop? predicting stopout in massive open online courses. *CoRR*, abs/1408.3382, 2014.

[24] N. Thai-Nghe, A. Busche, and L. Schmidt-Thieme. Improving academic performance prediction by dealing with class imbalance. In *Ninth International Conference on Intelligent Systems Design and Applications, ISDA 2009, Pisa, Italy , November 30-December 2, 2009*, 878–883, 2009.

[25] H. Vossensteyn, A. Kottmann, B. Jongbloed, and F. Kaiser. Drop-out and completion in higher education in europe executive summary. Technical report, European Commission, 2015.

[26] C. Wladis, A. C. Hachey, and K. M. Conway. An investigation of course-level factors as predictors of online STEM course outcomes. *Computers & Education*, 77:145–150, 2014.

[27] A. Wolff, Z. Zdrahal, D. Herrmannova, J. Kuzilek, and M. Hlosta. Developing predictive models for early detection of at-risk students on distance learning modules. In *Machine Learning and Learning Analytics workshop at LAK14, 24-28 March 2014, Indianapolis, Indiana, USA*, 4, 2014.

[28] C. Ye and G. Biswas. Early prediction of student dropout and performance in moocs using higher granularity temporal information. *Journal of Learning Analytics*, 1(3):169–172, 2014.