AmonAI: A Students Academic Performances Prediction System

Iffanice B. Houndayi, Vinasetan Ratheil Houndji(✉), Pierre Jérôme Zohou, and Eugène C. Ezin

Institut de Formation et de Recherche en Informatique (IFRI),
Université d'Abomey-Calavi (UAC), Abomey-Calavi, Benin
ratheil.houndji@uac.bj

Abstract. This paper presents a system, called AmonAI, that predicts the academic performances of students in the LMD system. The approach used allows to establish, for each of the teaching units of a given semester, some estimates of the students results. To achieve this, various machine learning techniques were used. In order to choose the best model for each teaching unit, we have tested 9 different algorithms offered by the Python Scikit-learn library to make predictions. The experiments were performed on data collected over two years at "Institut de Formation et de Recherche en Informatique (IFRI)" of University of Abomey-Calavi, Benin. The results obtained on the test data reveal that, on five of the nine teaching units for which the work was conducted, we obtain an F2-score of at least 75% for the classification and an RMSE of less than or equal to 2.93 for the regression. The solution therefore provides relatively good results with regard to the dataset used.

Keywords: Students performances prediction · Machine learning · Classification · Regression · Teaching unit · LMD

1 Introduction

The increasing use of ICTs in the different socio-economic fields has contributed to the generation of a large amount of data. The analysis of these data by humans can be a difficult task. Thus, several disciplines such as machine learning are involved in extracting knowledge or highlighting interesting structures from these data in order to solve problems or improve existing solutions. When one is interested in the field of education, from all the interactions and data produced, a large amount of information containing hidden patterns is also generated. To ensure that students are properly trained, it is important that they receive adequate support to improve and succeed in their studies. Unfortunately, several conditions (like huge number of students) make more difficult to monitor students; a situation that decreases their chances of success. This work is part of an initiative to reduce the failure rate of students. It will then dive into the application of machine learning techniques on the available data to make

R. Zitouni et al. (Eds.): AFRICOMM 2019, LNICST 311, pp. 212–218, 2020.
https://doi.org/10.1007/978-3-030-41593-8_16

the prediction of academic performances of the students in the LMD system. This paper presents a system that allows to anticipate their results in order to reduce their failure as much as possible (by taking appropriate decision). The system makes the prediction of the students academic performances through classification and regression in each teaching unit of a given semester and provide visualizations based on these predictions. We have developed a prototype for the "Institut de Formation et de Recherche en Informatique (IFRI)" of University of Abomey-Calavi, Benin. One of the challenge of this work is that such university schools do not store many social data (for example the distance between the student's house and the school, the fact that the student has an internet connection, etc.) that can be very useful here. On the other hand, our experiments show that any machine learning algorithm tested does not clearly dominates all others. Thus our system tests several machine learning algorithms (9 in this paper) to select the best one for each teaching unit.

This paper is organized as follows: Sect. 2 gives an overview of the related works; Sect. 3 presents our solution; Sect. 4 provides some experimental results got after applying our solution to available data and Sect. 5 concludes and gives possible directions for future works.

2 Related Works

Machine learning is nowadays widely used and its use is widespread in many fields, such as education, where obtaining a high success rate is a major challenge. Several researches were carried out in the sense of the prediction of the academic performances. In addition, these researches show that the use of machine learning in the field of predicting academic performance leads to good results.

A review on predicting students performance using Data Mining techniques was conducted at the School of Computer Science at Universiti Sains Malaysia [1]. It shows that the attributes frequently used by researchers are: the cumulative grade point average (CGPA), which is the most important input variable, internal evaluation (lab work, class queries, presence), student demographics (gender, age, family history and disability), external assessments (final exam score for a particular subject), extracurricular activities, secondary studies, social interaction, and psychometric factor (rarely used because it is based on qualitative data). A study conducted in April 2017 by Ali Daud et al. deals with the prediction of student performance (in terms of dropout: degree completed or dropped) using advanced learning techniques [3]. This research paper presents the prediction methods used, which use four different types of attributes, namely: family expenses, family income, student personal information and family assets. Another study carried out at the Tampere University in June 2017 by Murat Pojon addresses the theme of the use of machine learning to predict the performance of students, whose specific objective in this case is to measure the improvement made by feature engineering according to the performance of algorithms [2]. It focuses on linear regression, decision trees, and naive Bayes to make prediction of classification type. Better prediction results were obtained when feature engineering was applied. But the combination of method selection and feature engineering approaches provided the best results. Similar works have been done by other researchers including in the University of Minho in 2008 by Paulo Cortez and Alice Silva where the subject of academic prediction has been applied to high school students (specifically predict students results in mathematics and Portuguese) [4]. The results show that the students performances are strongly affected by their previous results. Interested readers may refer to [5–7] to see some other related works.

An important remark is that there is no algorithm that is suitable for any type of data. The method used is strongly conditioned by the structure and the content of the data. Since we do not have the wide range of features used in the previous works (extra-scholar, social data, etc.), we propose an adapted and contextualized solution to the data available at IFRI, UAC.

3 Our Solution

Our system, called `AmonAI`[1], predicts academic performances through a web platform. It allows to estimate students performances of a given semester by making predictions of the students results in each teaching unit of the concerned semester. These predictions are of two kinds, classification and regression. Classification means that the system predicts whether a student validates or not a teaching unit while regression means that students grades results in each teaching unit are anticipated.

The system of `AmonAI` is based on multiple classes. It contains the classes **User**, **Advanced User**, **Report**, **Semester** and **Analysis**. We present below the two important classes **Semester** and **Analysis**:

- the class **Semester** is used to record data for a semester. It is linked to the **User** class, which means that a Semester object has an **author** property of type **User**. When a semester is added with the training files (sample of previous semester data which will be used for inputs and sample of the current semester data which will be used for outputs/outcomes), it is possible for a user to generate the predictors of this semester, which will be used to generate the report of an analysis related to the concerned semester;
- the class **Analysis**, also linked to the class **User**, is used to configure the information relating to the analysis that the user wishes to perform. He/She specifies in particular the type of the analysis (classification or regression analysis), the file of the analysis, and the semester to which this analysis is related.

For the prediction phase, depending on the analysis to be performed by the user, an analysis file is specified. Then if the structure of this file matches with the one of the sample of previous semester data, a pre-processing is done in order to clearly identify the input variables that will be used for the prediction. Depending on the type of analysis, the predictors of the semester in which the

[1] `Amon` (in Fongbé) is a prediction of the oracle `Fâ`, AI stands for Artificial Intelligence.

analysis is related will make a prediction of performance for each student in the analysis file according to each semester teaching unit. The predicted outcomes for each teaching unit are **validated/non-validated** and a **score between 0 and 20** for respectively a binary classification analysis and a regression analysis.

Algorithm 1. Algorithm describing the performances prediction phase

```
Input: An instance $a$ of the Analysis class
Output: The performances predictions of all the students in the analysis file of $a$
begin
    if structure(a.analysisFile) = structure(a.basisSemester.trainingFilePreviousSem) then
        $students \leftarrow preprocessing(a.analysisFile)$;
        if $a.type$ = "Classification" then
            $predictors \leftarrow a.basisSemester.classificationPredictors$;
        else
            $predictors \leftarrow a.basisSemester.regressionPredictors$;
        end
        $predictions \leftarrow [\,]$;
        $listTeachingUnits \leftarrow a.basisSemester.listTeachingUnits$;
        for $i \leftarrow 0$ to $length(students) - 1$ do
            $prediction\_student \leftarrow [\,]$;
            for $j \leftarrow 0$ to $length(listTeachingUnits) - 1$ do
                $\hat{y} \leftarrow predictors[j].predict(students[i])$;
                $prediction\_student[j] \leftarrow \hat{y}$;
            end
            $predictions[i] \leftarrow prediction\_student$;
        end
        return $predictions$;
    else
        return ("Error! Analysis should be reconfigured");
    end
end
```

4 Experimental Results

4.1 Data and Algorithms Used

For the experiments, we have used the IFRI's data. At IFRI the cycle for bachelor degree consists of three (03) academic years, each with two semesters. The courses taught concern several teaching units subdivided in subjects (for example Mathematical Logic, C language, etc.). This work took into account the available data, which concerned those of the first year of bachelor in IT security and software engineering collected over two years (2016–2017 and 2017–2018). The prediction task was therefore performed on the second semester of the first year (not having relevant data to do so for the first semester) compared to which nine (09) of the ten (10) teaching units were taken into account (the teaching unit of Discipline being the one that has been isolated). Thus, as a basis for training phase, data on marks in first semester subjects and social data such as age and gender have been used. Finally, after pre-processing and isolation of irrelevant information, the data was collected in a dataset (with 258 instances) then separated into two parts using the 70/30 train-test split: 180 instances for the training and 78 instances for the tests.

Unlike in the previous studies we do not have lot of extra school data. We will so focus on applying multiple techniques in terms of algorithms. Thereby, before getting the best models for each teaching unit, the following algorithms were tested: Support Vector Machines (SVM), Decision Trees, Random Forest, Ridge regression (used specifically for regression), Logistic Regression (used specifically for classification), AdaBoost, Gradient Boosting Machine (GBM), k-Nearest Neigbors (KNN) and Feed Forward Neural Network (Multi-layer Perceptron: MLP).

4.2 Algorithms Evaluation

For the selection and evaluation of models, the "Training-Validation-Test" approach was used. The k-fold cross-validation method was performed on the training dataset for the selection of models and parameters. Thus, for the effective evaluation of the models, unknown data not having intervened in the training and validation phases were used: as previously mentioned a sample of 78 instances was used to make the tests.

With regard to classification, the null hypothesis is fixed to the fact that a student does not validate a teaching unit[2]. In order to detect as much as possible the cases of students who might not validate a teaching unit, it is preferable in this context to make a type II error, that is, to accept the null hypothesis whereas it's wrong. In this case, as an evaluation metric we have used the **F2-score** [8]. For regression, the metric used for the evaluation is the Root Mean Square Error (RMSE). Tables 1 and 2 show respectively the performance results (F2-scores and RMSEs) of the various algorithms after cross-validation and hyper-parameters tuning about the classification and regression tasks for each teaching unit.

Table 1. Summary of algorithms performances for classification (results rounded to 10^{-2} - best algorithms scores per teaching unit in bold - best scores per algorithm underlined)

F2-scores									
Algorithms	Teaching units (TUs)								
	TU1	TU2	TU3	TU4	TU5	TU6	TU7	TU8	TU9
Support Vector Machines	0,88	**0,87**	0,76	0,76	0,55	0,57	**0,60**	0,65	0,62
Decision Tree	0,85	0,70	0,65	0,68	0,28	0,40	0,57	0,68	0,47
Random Forest	0,81	0,83	0,74	0,72	0,51	**0,65**	0,53	**0,75**	0,54
Logistic Regression	0,80	0,80	**0,77**	**0,79**	0,53	0,59	0,57	0,68	**0,65**
AdaBoost	0,73	0,80	0,66	0,62	**0,57**	0,26	0,58	0,62	0,40
Gradient Boosting	0,91	0,83	0,54	0,13	0,53	0,34	0,38	0,1	0,41
KNN	0,83	0,79	0,44	0,56	0,47	0,38	0,31	0,40	0,40
Feed forward neural network	**0,92**	0,83	0,57	0,51	0,29	0,52	0,38	0,66	0,47

[2] The positive class is then "Non-validated".

Table 2. Summary of algorithms performances for regression (results rounded to 10^{-2} - minimum algorithms errors per teaching unit in **bold** - minimum errors per algorithm underlined)

RMSEs									
Algorithms	Teaching units (TUs)								
	TU1	TU2	TU3	TU4	TU5	TU6	TU7	TU8	TU9
Support Vector Machines	1,97	2,62	2,70	2,31	2,63	**2,89**	2,56	**2,93**	2,91
Decision Tree	2,41	3,10	4,34	2,90	2,73	3,81	3,73	3,51	3,44
Random Forest	**1,92**	**2,60**	2,51	2,23	2,58	2,91	**2,50**	3,04	2,85
Ridge Regression	2,11	2,90	3,04	**2,10**	**2,41**	2,92	2,71	3,28	2,84
AdaBoost	2,01	2,75	2,65	2,31	2,62	3,17	3,19	3,06	2,82
Gradient Boosting	2,02	2,69	**2,49**	2,37	2,74	2,93	2,65	3,10	**2,81**
KNN	2,30	2,76	2,52	2,46	2,85	3,01	2,62	3,51	2,94
Feed forward neural network	2,09	3,17	3,19	2,18	2,63	3,46	3,38	3,49	2,86

5 Conclusion and Perspectives

We have presented AmonAI, a system based on machine learning techniques that predicts students results in each teaching unit of a given semester. We have tested 9 different algorithms in order to choose the best one for each teaching unit. For the evaluation of the different algorithms which were tested, the metrics F2-score and RMSE were respectively used for classification and regression tasks. The different predictions are globally good with regard to our dataset. On 5 of 9 teaching units, the F2-score is $\geq$75% (classification) and the RMSE is $\leq$2.93 (regression) in all the teaching units.

For future works, it would be interesting to obtain a larger sample for training the algorithms (including other academic and extra-school data) because they contain several key aspects that were not considered in this work. In the same way, it would be important to perform more advanced pre-processing on the data. Finally, we would like to add a system of recommendations that will exploit the results from the predictive analysis to make suggestions.

References

1. Shahiri, A.M., Husain, W., Rashid, N.A.: A review on predicting student's performance using data mining techniques. School of Computer Sciences, Universiti Sains Malaysia (2015)
2. Pojon, M.: Using machine learning to predict student performance. University of Tampere (2017)
3. Daud, A., Aljohani, N.R., Abbasi, R.A., Lytras, M.D., Abbas, F., Alowibdi, J.S.: Predicting student performance using advanced learning analytics. In: Proceedings of the 26th International Conference on World Wide Web Companion, pp. 415–421 (2017)
4. Cortez, P., Silva, A.: Using data mining to predict secondary school student performance. University of Minho (2008)

5. Meier, Y., Xu, J., Atan, O., van der Schaar, M.: Predicting grades. IEEE Trans. Signal Process. **64**(4), 959–972 (2016)
6. Agrawal, H., Mavani, H.: Student performance prediction using machine learning. Int. J. Eng. Res. Technol. **4**(03), 111–113 (2015)
7. Github: Student Performance Prediction. https://github.com/sachanganesh/student-performance-prediction. Accessed 9 Jan 2019
8. Clusteval: Integrative Clustering Evaluation Framework F2-Score. https://clusteval.sdu.dk/1/clustering_quality_measures/5. Accessed 7 Nov 2018