Can the Paths of Successful Students Help Other Students With Their Course Enrollments?

Kerstin Wagner
BHT[*] Germany
kerstin.wagner@bht-berlin.de

Agathe Merceron
BHT[*] Germany
merceron@bht-berlin.de

Petra Sauer
BHT[*] Germany
sauer@bht-berlin.de

Niels Pinkwart
DFKI[†] Germany
niels.pinkwart@dfki.de

ABSTRACT

In this paper, we present an extended evaluation of a course recommender system designed to support students who struggle in the first semesters of their studies and are at risk of dropping out. The system, which was developed in earlier work using a student-centered design and which is based on the explainable k-nearest neighbor algorithm, recommends a set of courses that have been passed by the majority of the student's nearest neighbors who have completed their studies. The present evaluation is based on the data of students from three different study programs. One result is that the recommendations do lower the dropout risk. We also discovered that while the recommended courses differed from those taken by students who dropped out, they matched quite well with courses taken by students who completed the degree program. Although the course recommender system targets primarily students at risk, students doing well could use it. Furthermore, we found that the number of recommended courses for struggling students is less than the number of courses they actually enrolled in. This suggests that the recommendations given indicate a different and hopefully feasible path through the study program for students at risk of dropping out.

Keywords

Course recommender system, nearest neighbors, explainability, user-centered design, dropout prediction

1. INTRODUCTION

In the last decades, universities worldwide have changed a lot. They offer a wider range of degree programs and courses and welcome more students from diverse cultural backgrounds. Further, teaching and learning at school differs from teaching and learning at university. Some students cope well and keep the same academic performance level at university as at school. Others struggle, perform worse, and might become at risk of dropping out.

The preliminary exploration of our data has shown, that most of the students drop out during the first three semesters of their studies. Therefore, the course recommendations proposed in this work focus on supporting struggling students after their 1st and 2nd semesters. The final goal in developing such a system is to integrate it into novel facilities that universities may set up to support their diverse students better.

At the beginning of each semester in Germany, students must decide which courses to enroll in. When entering university directly after high school for their 1st semester, most of them decide to enroll in exactly the courses planned in the study handbook. The decision becomes more difficult when they fail courses in their 1st semester and should choose the courses to enroll in their 2nd semester: should they repeat right away the courses they failed? Which courses planned in the 2nd semester in the study handbook should they take? Should they reduce the number of courses they enroll in to have a better chance of passing them all? Should they take more courses to compensate for the courses they failed? The study handbook does not help answer these questions.

Previous research has shown that most students rely on friends and acquaintances as one source of information when deciding which courses to enroll in [19]. Further, students wish to have explanations if courses are recommended to them. The present recommender system supports students in choosing which courses to take before the semester begins and is based on the explainable algorithm k-nearest neighbors (KNN). It recommends to students the set of courses that the majority of their nearest neighbors, who successfully graduated, have passed.

Nearest neighbors are students who, at the same stage in their studies, have failed or passed almost the same courses with the same or very similar grades. The system does not recommend top n courses as other systems do, e.g. [10, 12, 14, 15]. Rather, it recommends an optimal set of courses, and we assume that a student should be able to pass all the courses of that set. Because the recommendations are driven by past records of students who graduated, we also pose the hypothesis that students following the recommendations

[*] Berliner Hochschule für Technik

[†] Deutsches Forschungszentrum für Künstliche Intelligenz

should have a lower risk of dropping out. Using historical data, we evaluated the recommendations given after the 1st and 2nd semester. Although the recommendations are designed to support struggling students, every student should have access to them. The recommendations should show a different, more academically successful way of studying for struggling students and therefore differ from the courses that they pass or enroll in.

More precisely, this paper addresses the following research questions:

1. Do the recommendations lower the risk of dropping out?
2. How large is the intersection between the set of courses recommended and the set of courses a student has passed?
3. a) How many courses are recommended and b) does this number differ from the number of courses passed and enrolled in by students?

This work builds on our previous work [20] by using a larger dataset with three different study programs instead of one to answer research questions 1 and 2, and by adding research question 3 to further investigate the provided recommendations. For all three questions, it is relevant whether there is a difference between students with difficulties and students with good performance as well as between study programs and semesters.

The paper is organized as follows. The next section describes related works. In the third section, we present our data, and in the fourth section our methodology. The results and their discussion are presented in section 5. In section 6, we describe a preliminary evaluation with students. The last section concludes the paper and presents future works. To make this paper self-contained, sections 3 and 4 repeat some descriptions and explanations already presented [20].

2. RELATED WORK

Dropout Prediction. Since our work aims to support students at risk of dropping out, it is necessary for us to be able to assess students' risk. Researchers have used various data sources, representations, and algorithms to address the task of predicting dropout. Academic performance data quite often form the basis; adding demographic data does not inherently lead to better results [2] but has been done for example in [1, 2, 9]. The data can be used as is as features or aggregated into new features. In terms of the algorithms used for dropout prediction, they range from simple, interpretable models such as decision trees, logistic regression, and KNN [1, 2, 9, 21] to black-box approaches like AdaBoost, random forests, and neural networks [1, 2, 11] — there is no algorithm that performs best in all contexts. Because the current study examines the impact of course recommendations on predicted risk, we only use courses and their grades as features when performing dropout prediction in section 4.2.

Course Recommendations. Various approaches to course recommendation have been explored in recent years. Urdaneta-Ponte et al. provided an overview of 98 studies published between 2015 and 2020 and related to recommender systems in education [18]. They answered the questions, among others, about what items were recommended and for whom the recommendations were intended. Course recommendations were found to be the second most common research focus, with 33 studies after learning resources, and 25 of these papers targeted students. Ma et al. first conducted a survey to identify the factors that influence course choice [10]. Based on this, they developed a hybrid recommender system that integrates the aspects of interest, grades, and time into the recommendations. The approach was evaluated with a dataset containing the results of 2,366 students from 5 years and from 12 departments. They obtained the best results in terms of recall when all aspects are included but with different weights. Morsy and Karypis analyzed their approaches to recommend courses in terms of their impact on students' grades [12]. Based on a dataset that includes 23 majors with at least 500 graduated students from 16 years, they aim to improve grades in the following semester without recommending easy courses only. Elbadrawy and Karypis investigated how various student and course groupings affect grade prediction and course recommendation [6]. The objective was to make the most accurate projections possible. Around 60,000 students and 565 majors were included in the dataset. The list of courses from which recommendations were derived was pre-filtered by major and student level. This limitation is comparable to our scenario, in which students choose courses depending on their study program. None of these works has the aim of supporting struggling students when enrolling in courses.

Our contribution. The idea of building a recommender system to support struggling students in their course enrollment, based on the paths of fellow students with the potential of providing explanations came out of the insights gained from a semi-structured group conversation with 25 students [19]. We propose a novel, thorough approach to evaluate such a recommender system that includes the following characteristics:

- Studies have shown that course recommendations can have an impact on students' performance. However, students at-risk were not in focus. We employ a two-step dropout risk prediction to determine whether the recommendations reduce dropout risk.
- We recommend a set of courses, not top n courses; therefore we evaluate not only that the passed courses contain the recommended courses — similar to other evaluations [6, 10, 12] — but also that the recommended courses contain the courses students have passed using F_1 score.
- We evaluate whether the number of courses is adequate.

3. DATA

Data from three six-semester bachelor programs at a medium-sized German university were used to develop and evaluate the course recommender system: Architecture (AR), Computer Science and Media (CM), and Print and Media Technology (PT). These three programs differ not only in their topic but also in the number of students enrolled. The initial dataset included 3,475 students who began their studies between the winter semester of 2012 and the summer semester of 2019. We only used data about academic performance: students' course results from the first three semesters accounted for 45,959 records of information about course enrollments and exam results over the mentioned period. The

Table 1: Number of students by program P (AR, CM, PT), train and test data set (Type), and student status (D, G). The proportion of dropouts in the test dataset is used as risk indicator (Risk).

P	Type	D	G	Sum	Risk
AR	Train	91	371	462	0.197
	Test	43	73	116	0.371
CM	Train	154	267	421	0.366
	Test	67	39	106	0.632
PT	Train	37	171	208	0.178
	Test	21	32	53	0.396
AR + CM + PT		413	953	1,366	0.302

grading scale is [1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0, 5.0], with 1.0 being the best, 4.0 being the worst (just passed), and 5.0 means fail. Students may enroll in courses without taking the exam. In this case, they do not receive a grade, but the enrollment is recorded. To graduate, students must pass all mandatory courses as well as a program-specific number of elective courses. The study handbook includes a suggested course schedule for the six semesters, which students may or may not follow. At any time in their studies, students are allowed to choose courses from all offered courses.

Outliers. We removed three types of students: A) outliers in terms of the number of passed courses based on the interquartile range. Indeed, students can receive credit for courses completed in previous study programs; in our data, these credits are not distinguishable from credits earned by enrolling in and passing a course but they may result in a large number of courses passed, far more than anticipated in the study handbook. We remove these outliers because they might impact negatively dropout prediction [13]. B) students who were still studying at the time of data collection since they can not be used to predict the risk of dropping out. C) students without at least one record (passed, failed, or enrolled but have not taken the exam) in each of the first three semesters.

Datasets. The final dataset included 1,366 students who either graduated ("graduates", status G) or dropped out ("dropouts", status D). For the programs AR and CM, we had similarly sized data sets with 578 and 527 students, but only 261 students for the PT program because it has fewer students, see programs AR, CM, and PT, rows train and test column Sum in Table 1. For dropout risk prediction, described later in section 4.2, the data sets were sorted by the start of the study and split into 80% training data, row train in Table 1, and 20% test data, row test in Table 1, so that prediction evaluation was done based on students who started their studies last. We call the proportion of dropouts in each data set "dropout risk", see column Risk in Table 1. For example, the dropout risk of the train set of the program Architecture AR is 0.20= 91/462. Table 2 provides an overview of the number of courses students enroll and pass on average, the difference between the number of courses enrolled and passed, and the average grade based on courses passed and failed, by program, semester, and student status. For example, students in program AR who dropped out in the first semester enrolled in 4.9 courses but passed 3.2 courses on average, and got an average grade of 2.8, whereas students who graduated enrolled in 5.0 courses and passed 4.7 courses on average, and got an average grade of 2.1. One notices that students with status D pass fewer courses per semester and receive lower grades.

Table 2: Academic performance overview by program and semester (PS), and student status (D, G): mean number of courses enrolled in (MeanE), mean number of courses passed (MeanP), difference (Diff) between MeanE and MeanP, and mean grade (MeanGr).

	MeanE		MeanP		Diff		MeanGr	
PS	D	G	D	G	D	G	D	G
AR1	4.9	5.0	3.2	4.7	1.7	0.3	2.8	2.1
AR2	5.5	5.8	3.0	5.1	2.5	0.8	3.0	2.3
AR3	5.1	5.9	1.9	5.4	3.2	0.5	3.2	2.2
CM1	4.9	5.1	2.9	4.8	2.0	0.3	3.0	2.1
CM2	5.2	5.8	2.1	5.0	3.1	0.8	3.0	2.3
CM3	4.7	5.8	1.3	5.0	3.5	0.9	3.2	2.1
PT1	5.8	6.0	4.3	5.8	1.5	0.3	2.5	2.0
PT2	5.7	5.5	2.5	4.9	3.2	0.6	2.9	1.9
PT3	6.1	6.4	2.3	5.5	3.8	0.9	3.1	2.0

Missing values. For the algorithms used for the recommendations and dropout predictions, we had to deal with missing values. If students enrolled in a course but did not take the exam, a 6.0 was imputed; if they were not enrolled at all, a 7.0 was imputed. This means that not enrolling (7.0) is penalized more than enrolling but not taking the exam (6.0).

Data representation. Each student is represented by a vector of grades. It is possible for a student to, for example, enroll in a course in the 1st semester and not take the exam, then enroll and fail the exam in the next semester and enroll again and pass the exam in the following semester. In this case, a student has three different records for the same course in three different semesters. In our opinion, not only the final grade with which a course was passed is relevant, so we include the entire history of a student's academic performance in the vector. Table 3 shows the vector representation of six students for their three first semesters of study. Note that the courses where all students have the value 7.0 are omitted. Students 0, 3, and 5 enrolled in the course M03 without taking the exam in semester 1 (value 6.0), students 0 and 3 did the same in semester 2 but did not enroll in semester 3 (value 7.0), while student 5 did the opposite; students 1, 2, and 4 passed M03 in semester 1.

Table 3: Example of a course recommendation for one student with five neighbors for the 3rd semester. The columns show the semesters (1, 2, 3) and the courses the students were enrolled in, e.g. M01, M02. Row 0 represents the student who receives a recommendation, and rows 1 to 5 represent the student's five nearest neighbors. The recommended courses are highlighted in blue. The cells show their grades; 6.0 and 7.0, colored in gray, are imputed for missing values. The actual grades of student 0 in semester 3 are given for comparison and highlighted in italic.

S	1					2							3								
	M01	**M02**	**M03**	**M04**	**M05**	**M03**	**M08**	**M09**	**M10**	**M11**	**M12**	**M13**	**M03**	**M10**	**M11**	**M14**	**M15**	**M16**	**M17**	**M18**	**M19**
0	1.3	2.7	6.0	2.3	2.0	6.0	1.3	2.3	7.0	2.7	2.7	2.7	*7.0*	*2.3*	*7.0*	*3.0*	*3.0*	*2.3*	*2.7*	*2.0*	*7.0*
1	1.7	2.7	2.0	2.7	2.3	7.0	2.0	2.7	6.0	4.0	2.3	2.3	7.0	7.0	7.0	1.3	1.3	1.7	3.0	1.7	4.0
2	2.0	1.7	2.3	2.3	1.7	7.0	1.3	2.3	7.0	3.0	2.3	5.0	7.0	7.0	7.0	2.0	2.0	7.0	2.0	2.7	1.3
3	2.3	2.0	6.0	2.7	2.0	6.0	1.3	1.7	6.0	6.0	2.0	2.0	7.0	7.0	6.0	2.7	1.7	1.7	3.0	2.0	2.7
4	2.3	2.3	3.0	2.3	1.0	7.0	2.7	2.0	6.0	1.3	2.3	2.3	7.0	7.0	7.0	1.7	1.7	2.0	1.0	1.3	3.7
5	1.3	3.0	6.0	2.3	3.0	7.0	1.7	6.0	6.0	2.3	2.7	2.3	6.0	7.0	7.0	3.0	5.0	2.7	6.0	5.0	7.0

4. METHODOLOGY

In this section, we present first the course recommender system. Then we explain the two-step dropout prediction and how we optimized the models. Finally, we describe the evaluation of the prediction models for RQ1 and the course recommendations for RQ2 and RQ3. In our case, since many students drop out after the first or second semester, we consider the recommendations and dropout predictions for the second and third semesters. For each research question, we look at subgroups by program, semester, and student status.

4.1 Course Recommendations

The course recommender system is based on a KNN classifier: given a student represented by a vector of grades at the end of semester t, the majority votes of his/her neighbors classify a course as "passed" and accordingly recommended for the following semester $t + 1$. KNN has the advantage that the neighbors can be calculated only once and on their basis, the classification can be made for all courses. Since we considered all courses passed by any student in semester $t + 1$, we got two sets: "recommended" and "not recommended". Given the possibility to recommend a course that the student being observed has already passed, we removed this course from the recommendation if necessary. We recommended courses for all 1,366 students to have the largest possible database to evaluate the recommendations.

Parameters. To avoid a tie in majority voting, we used only uneven k and tested our approach with k from 1 to 25. Additionally, we selected the Euclidean Distance as distance metric for calculating the distances between the students.

Risk reducing approach and baseline. To ensure that courses are recommended that reduce dropout risk, we included in our approach only neighbors who graduated from the program. As a baseline for comparison, we used also all neighbors, which means including neighbors who dropped out, to generate course recommendations. We expected that the recommendations differ depending on the neighbor type and that the recommendations based on graduated students, but not necessarily the recommendations generated with all students, reduce the risk of dropping out. In the following, we distinguish the two neighbor types with AN (all neighbors) and GN (neighbors who graduated).

Example. Table 3 shows the data used to calculate the neighbors and to recommend courses to student 0 for the 3rd semester. The actual grades — or imputed values 6.0 and 7.0 if grades were missed — for relevant courses (M01 to M19) are shown for each semester. Semesters 1 and 2 are the previous semester on which the distance calculation is based. Semester 3 covers the course recommendations. A course is passed if the grade lies between 1.0 and 4.0. M10 was not recommended to student 0 in this example but student 0 passed it in semester 3, M19 was recommended because four of five neighbors passed it but student 0 did not enroll in it.

4.2 Dropout Risk Prediction

A dropout prediction was performed using the following two steps: 1) A model was trained to predict the two classes: dropout (D) or graduate (G) based on actual enrollment and exam information; 2) The model from step 1 was used again to predict dropout or graduation after the calculated recommendations replaced the actual enrollment and exam information. We call "dropout risk" the proportion of students in the test set who are predicted to drop out in this prediction task. To determine whether or not the recommended courses help to lower the dropout risk, we compare the predicted dropout risk from step 1 (P_1) with the predicted dropout risk from step 2 (P_2). The goal is for P_2 to be less than P_1.

4.2.1 Step 1

Feature set. As investigated by Manrique et al. [11], there are several ways to select a feature set for dropout prediction and no way works better than the others in all contexts. Because we want to measure the impact of our recommendations on dropout prediction, we use the courses taken by students as features; the values of the features are the grades.

Model training. To detect a change in the dropout risk, the models should be as accurate as possible which we aimed to achieve through two approaches: A) train different types of algorithms, and B) use different approaches for optimization. For all cases, the datasets were sorted by students' study start and then split into 80% training data and 20% test data, so that risk prediction is done for students who started their studies last. As can be seen in Table 1, the proportion of dropouts is higher in the test set than in the training set because it usually takes six semesters to know whether a student will graduate whereas many students drop out of their studies much earlier. We trained models for each program (AR, CM, PT) and semesters $t = 2$ and $t = 3$ with actual grades and used the best models to evaluate a change in dropout risk in step 2).

Algorithms. We trained the following algorithms in Python using scikit-learn: decision tree (DT), lasso (L, penalty=l1, solver=liblinear), logistic regression (LR, penalty=none, solver=lbfgs), k-nearest neighbors (KNN), random forest (RF), support vector machine with different kernels (SV: rbf, LSV: linear, PSV: poly).

Optimization. Using our experience in [20], we simply use the scikit-learn default hyperparameter settings, except the settings to obtain a certain algorithm as mentioned above, in combination with the following list of algorithm-independent parameters. i) Feature selection by cut-off (CO): we removed courses with too few grades and tried 1 and 5 as a minimum number of grades to retain a course; a value that is too high may result in the removal of recommended courses and thus would not be included in the dropout prediction. ii) Training data balancing (BAL): we used two common techniques: Synthetic Minority Oversampling Technique (SMOTE) [4] and RandomOverSampler (ROS). Both implementations are from imbalanced-learn, a Python library. iii) Decision threshold moving (DTM): Usually, a classifier decides for the positive class at a probability greater or equal to 0.5, but in case of imbalanced data, it may be helpful to adjust this threshold, so we checked additionally to 0.5 values between 0.3 and 0.6 in 0.05 steps. Lower and higher values did not lead to better results.

Evaluating the model performance. To emphasize that both correct dropouts and correct graduates are important for dropout risk prediction, we evaluated the models based on the test data using the Balanced Accuracy metric (BACC), defined as the mean of the recall for class 1 (dropout), also known as true positive rate, and recall for class 0 (graduate), also known as true negative rate: $BACC = (TP/P + TN/N)/2$.

4.2.2 Step 2

In the second step, we used the best model by BACC from step 1 for each program and the semesters $t = 2$ and $t = 3$ to predict dropout. The dropout prediction for $t = 2$ used the actual grades of the 1st semester and the recommendations for the 2nd semester, and the prediction for $t = 3$ used the actual grades of the 1st and 2nd semesters and the recommendations for the 3rd semester. For the recommendations, we assumed that the student can pass the recommended courses. For student 0 in Table 3, courses from M14 to M19 are recommended and we assume that s/he will pass all these courses in semester 3. If we had an actual grade in the records for that student and a recommended course, we used it. If not, we predicted a grade by imputing the average of two medians: the median of all the grades that we know about from the student and the median of the historical grades for that course. This imputation rests on the strong assumption that underpins our recommendations: the majority vote of the k nearest neighbors yields a set of courses that a student can pass. We evaluated this grade prediction using the known actual grades and obtained a Root Mean Square Error (RMSE) of 0.634, which is comparable with RMSE scores from 0.63 to 0.73 to other studies in that field [6, 16]. Consider again student 0 in Table 3. In addition to the courses from semesters 1 and 2, M10 and M14 to M18 from the third semester were used for prediction in step 1, and M14 to M19 from the third semester were used in step 2 with a predicted grade for M19.

Table 4: Structure of the confusion matrix for recommendation evaluation for one student.

	Predicted positive	**Predicted negative**	**Totals**
Actual positive	Passed and recommended True positive TP	Passed but not recommended False negative FN	**Passed P**
Actual negative	Not passed but recommended False positive FP	Not passed and not recommended True negative TN	**Not passed**
Totals	**Recommended**	**Not recommended**	**All courses**

4.3 Evaluation

4.3.1 RQ1 Evaluation

To answer the question ”Do the recommendations lower the risk of dropping out?” in section 5.1, we compare the dropout risk, i.e. the proportion of students who are predicted to drop out, based on the predictions from step 2 (P_2) with those from step 1 (P_1). We also distinguish the neighbor types for step 2: P_{2AN} corresponds to the step 2 dropout prediction using the courses recommendations based on all neighbors (baseline) while P_{2GN} uses the recommendations based on graduate neighbors.

4.3.2 RQ2 Evaluation

Since the course recommendations are for each course a binary classification problem, we employ a confusion matrix for each student (Table 4) to answer the question ”How large is the intersection between the set of courses recommended and the set of courses a student has passed?” in section 5.2. We evaluate the recommendation for semester $t + 1$ for each student as follows: a course recommended and actually passed is a true positive (TP), a course recommended and actually not passed is a false positive (FP), a course not recommended but passed is a false negative (FN), and a course not recommended and not passed is a true negative (TN).

To evaluate a set of recommended courses, it’s important to measure both recall (whether passed courses include recommended courses) and precision (whether recommended courses include passed courses). We chose the F_1 score to evaluate courses’ intersections since the F_1 score ignores TN, which is in our context always a high value and thus does not serve our needs. The score ranges from 0 to 1 with 1 indicating perfect classification (recall=1 and precision=1) and 0 indicating perfect misclassification (recall=0 or precision=0). The calculation is as follows: $F_1 = 2 \cdot TP/(2 \cdot TP + FP + FN)$.

Further, we provide the recall which is in our case TP/P and equivalent to recall@ns, the percentage of recommended courses based on the number of courses taken by student s to enable comparison with similar work [10, 17]. Recall@n would fix the number of recommended courses at n [6, 14] and is not applicable in our case since we do not rank the recommendations and may also recommend less than n courses. Looking at the recommendations for student 0 in Table 3, the courses M14 to M18 are TP, M10 is FN, M19 is FP, and all the other 29 — here not shown — courses are TN. $F_1 = 2 \cdot 5/(2 \cdot 5 + 1 + 1) = 0.8\bar{3}$. $Recall = 5/6 = 0.8\bar{3}$. We aggregate the results as mean F_1 for both neighbor types and mean recall for neighbor type GN of all students grouped by student status, type of neighbors, program, and semester to compare the scores of the subgroups.

4.3.3 RQ3 Evaluation

To answer the question "a) How many courses are recommended and b) does this number differ from the number of courses passed and enrolled in by students?" in section 5.3, we look first at the number of courses recommended for semester $t + 1$. Using a horizontal barplot, we visualize the distribution of students by the number of recommended courses. To analyze why some students get no or only a few recommendations, we describe the relationship between the number of recommended courses and the distance of students to their neighbors. Using a scatterplot, we visualize the mean distance of a student to its neighbors in relation to the number of recommended courses. Second, we calculate the median difference between the number of courses recommended and courses passed, and the median difference between the number of courses recommended and courses enrolled. This may yield a difference in the number of courses students pass or enroll in than recommended by the system, depending on the subgroup.

5. RESULTS AND DISCUSSION

In this section, we first present the dropout prediction models and the changes in dropout risk based on the two-step prediction (RQ1). This includes identifying an appropriate value for k, the number of neighbors, that we use for the in-depth analysis of the course recommendations regarding the intersection (RQ2), and the number of courses (RQ3).

5.1 Dropout Risk

5.1.1 Dropout Prediction Models

Step 1 prediction. We selected the models — trained with actual exam and enrollment data — with the highest BACC for each program and semester (Table 5). They differ regarding their algorithm-independent parameters. We obtain P_1 as the step 1 dropout risk, i.e., the proportion of students from the test set predicted to drop out, that we compare later with the step 2 dropout risk P_2.

Example CM2. The support vector classifier (column C) achieved the best BACC when removing all courses that do not have at least one grade (column CO) resulting in 36 courses or features (column F); the decision threshold (column DTM) is 0.3, which means that students are predicted to drop out already at a 30% probability; the training set was not balanced (column BAL). Compared to the actual risk in the test data (0.632, Table 1 row CM > test), the predicted risk is lower (0.557).

Table 5: Best step 1 dropout prediction models for programs and semester (PS) regarding balanced accuracy (BACC) including their corresponding recall (REC), the classifier used (C), the number of used features (F), optimized parameters (CO, DTM, BAL), and the proportion of students of the test set who are predicted to drop out ($\mathbf{P_1}$).

PS	C	F	CO	DTM	BAL	P_1	BACC	REC
AR2	RF	38	0	0.35	SMOTE	0.353	0.866	0.814
AR3	RF	32	4	0.45	ROS	0.336	0.935	0.884
CM2	SV	36	1	0.30	None	0.557	0.920	0.866
CM3	RF	74	0	0.45	SMOTE	0.566	0.927	0.881
PT2	LSV	16	3	0.30	SMOTE	0.358	0.913	0.857
PT3	LSV	47	3	0.30	SMOTE	0.396	0.882	0.857

The best models have been obtained when the training data is balanced except for program CM and semester 2. The predicted dropout risk P_1 is lower in all cases than the actual dropout risk, see column Risk for the test set in 1, as we have observed for CM2, except for PT3 where it is equal. This means that our models tend to be optimistic and predict as graduate some students who dropped out.

5.1.2 Changes in Dropout Risk

Using the best models shown in Table 5, we performed the step 2 prediction using the recommendations.

Selecting an appropriate value for *k*. The set of recommended courses is critical for the step 2 prediction and depends on the number of neighbors k. Unfortunately, our research has shown that there is no value of k that generates an optimal set of courses for all three study programs and semesters and the two kinds of students: those who dropped out and those who graduated. Two values, $k = 3$ and $k = 5$, emerge as optimal or near-optimal and as never bad. The neighbors provide students with examples of how fellow students have enrolled and passed courses in their studies; this is one support that our students are looking for when they enroll [19]. Acknowledging this wish, matching the number of similar people used in the interviews by Du et al. [5], and in order to provide students with a variety of paths through their studies that are close to their own path, we choose $k = 5$ for further analysis in this work.

Step 2 prediction. Table 6 shows three proportions of students who are predicted as dropouts using the recommendations of five neighbors: P_1 from step 1, P_{2GN} based on neighbors who graduated, and P_{2AN} based on all neighbors. We distinguish the predicted dropout risk by student status, D or G, for a better overview of how the models perform.

Example CM2. Considering students who actually dropped out (D), 86.6% are predicted to drop out in step 1, 77.6% in step 2 using recommendations calculated with all neighbors

Table 6: Mean predicted dropout risk in step 1 ($\mathbf{P_1}$) and based on five neighbors and both neighbor types (AN, GN) in step 2 ($\mathbf{P_2}$) by student status (D, G), program and semester (PS). $\mathbf{P_{2GN}}$-$\mathbf{P_1}$ gives the corresponding change.

ST	PS	$\mathbf{P_1}$	$\mathbf{P_{2AN}}$	$\mathbf{P_{2GN}}$	$\mathbf{P_{2GN}}$-$\mathbf{P_1}$
D	**AR2**	0.814	0.674	0.558	-0.256
	AR3	0.884	0.721	0.279	-0.605
	CM2	0.866	0.776	0.716	-0.149
	CM3	0.881	0.821	0.716	-0.164
	PT2	0.857	0.619	0.619	-0.238
	PT3	0.857	0.905	0.810	-0.048
G	**AR2**	0.082	0.014	0.027	-0.055
	AR3	0.014	0.041	0.000	-0.014
	CM2	0.026	0.051	0.051	0.026
	CM3	0.026	0.051	0.026	0.000
	PT2	0.031	0.406	0.312	0.281
	PT3	0.094	0.250	0.188	0.094

(AN), and 71.6% using recommendations calculated with neighbors who graduated (GN). Looking at students who actually graduated (G), 2.6% are predicted to drop out in step 1, 5.1% in step 2 using recommendations calculated with all neighbors, and also 5.1% using recommendations calculated only with students who graduated. Thus, if we use the course recommendations and assume that these exact courses are passed, the risk decreases by 14.9% for actual dropouts and increases by 2.6% for actual graduates. Based on the size of the test dataset (Table 1), this means in absolute numbers: of 67 dropouts, 10 more students are predicted to graduate and of the 39 graduates, one more student is predicted to drop out compared to the step 1 prediction.

5.1.3 RQ1 Findings and Discussion

The question "Do the recommendations lower the risk of dropping out?" can be answered with yes, our approach lowers the dropout risk in most cases and we explore the risk reduction scores from different perspectives more precisely:

Graduates and dropouts. As we analyze Table 6, we expect the values in column P_{2GN} to be equal to or smaller than those in column P_1, and this holds true for students with status D, who are the primary focus of our recommendations. Additionally, for the AR program, we observe the same pattern for students with status G. However, for the program CM semester 2 and the program PT, the values in column P_{2GN} are higher than those in column P_1 , specifically for the graduated students. A glance at Table 1 reveals that the number of students with status G is small in the test set of CM2, while the program PT has a smaller number of students overall than the other two programs. This could explain these somewhat negative results, particularly for the PT program.

AN-based and GN-based recommendations. Comparing column P_{2AN} of Table 6 with column P_1, one notices that the values are everywhere smaller or equal in column P_{2AN} for the students with status D, except for PT3. This is less true for the students with status G. Comparing column P_{2AN} with column P_{2GN}, one notices that the values in column P_{2GN} are everywhere smaller or equal, except for the students with status G in AR2. These results indicate that calculating the recommendations by choosing the neighbors among all students could already be helpful. They also confirm that choosing neighbors among the students who graduated gives better results.

2nd and 3rd semester. Looking at the column $P_{2GN} - P_1$, we expect all values for the students with status G to be small, as not many students who graduated are predicted to drop out; one notices the small value -0.048 in PT3 for the students with status D. We conjecture that this is due to the high number of elective courses proposed in semester 3 of this study program. As students can freely choose five courses from six among a list of about 25 courses, it is more difficult for the algorithm to calculate accurate recommendations.

Overall, the results show that students who dropped out will benefit from enrolling and passing the courses recommended to them, above all when the recommendations are calculated with neighbors who have graduated. The assumption that students will pass the courses recommended to them sounds strong. However, as we shall see in section 5.3, the number of recommended courses is on average one course less than the number of enrolled courses. Focusing on fewer courses, as the recommendations suggest to them, might be helpful.

The findings indicate that the utilization of machine learning algorithms for assessment purposes may be constrained in scenarios where the student population is limited, particularly in the context of degree programs CM and PT with a small number of students possessing status G. The outcomes generated may not be reliable due to the small sample size. Additionally, the study reveals a limitation in recommendations based on nearest neighbors when the degree program is configured with a substantial number of elective courses, such as in the third semester of program PT. Therefore, relying on such recommendations may not be suitable in this particular scenario.

5.2 Courses' Intersection

We evaluate how the set of recommended courses calculated with five neighbors intersects with the set of courses students have passed using the means of the individual F_1 scores and recall (Table 7). To better distinguish for which student groups the recommendations better align with actual courses passed, the results are grouped by program and semester (PS), student status (D, G), and type of neighbors (AN, GN). Note that recall is shown when recommendations are calculated with neighbors from the set GN.

Example CM2. The F_1 score for students who actually dropped out (D) is 0.328 for recommendations based on all neighbors (AN) and 0.397 for recommendations based only on neighbors who graduated (GN). Looking at students who graduated (G), the F_1 score is much higher, 0.824 for rec-

Table 7: Mean F_1 score for neighbor types (AN, GN) and mean recall for neighbor type GN by student status (D, G), program, and semester (PS).

	$\mathbf{F_{1AN}}$		$\mathbf{F_{1GN}}$		$\mathbf{Recall_{GN}}$	
PS	**D**	**G**	**D**	**G**	**D**	**G**
AR2	0.481	0.854	0.521	0.871	0.649	0.925
AR3	0.279	0.817	0.305	0.842	0.417	0.875
CM2	0.328	0.824	0.397	0.851	0.498	0.895
CM3	0.130	0.711	0.159	0.755	0.187	0.788
PT2	0.511	0.837	0.528	0.828	0.651	0.844
PT3	0.112	0.335	0.156	0.356	0.140	0.284

ommendations based on all neighbors (AN) and 0.851 for recommendations based on neighbors who graduated (GN). Recall is 0.498 for students with status D and 0.895, again much higher, for students with status G.

5.2.1 RQ2 Findings and Discussion

We look at the question ”How large is the intersection between the set of courses recommended and the set of courses a student has passed?” from different perspectives.

Graduates and dropouts. The recommendations should show another, more promising way of studying to students who are struggling while they should not disturb students who are doing well. Thus, we expect the F_1 score and recall to be much higher for students with status G than for students with status D. We consider only the two columns GN on the right of Table 7 in the remainder of this section, namely recommendations calculated using neighbors who graduated as they gave the best F_1 results, which means that overall, graduate neighbors recommend better the courses that the students have actually passed. The column F_{1AN} is shown for the seek of completeness. As expected, the mean F_{1GN} score and recall are always higher for students with status G than for students with status D. F_{1GN} is higher than 82% in four cases and 75.5% in one case. Recall is always higher than 78%. This means that the recommended courses reflect quite well how these students study. An exception is program PT and semester 3. This might be due to the high number of elective courses offered by that program in semester 3. Of the 26 courses recommended to at least one student and also used in dropout prediction, only one is mandatory; the other 25 are electives. For students with status D, the mean F_{1GN} score tends to be low, around 52% in two cases and below 40% in the other cases.

2nd and 3rd semester. The mean F_1 score and the mean recall are higher in all cases for the 2nd semester than for the 3rd semester. The higher the semesters, the more the courses students pass drift apart. On the one hand, this makes it more difficult to find close neighbors, and on the other hand, it makes the recommendation itself more difficult: the neighbors sometimes disagree and have passed too many different courses, which means that no majority can be found for many courses and these courses are not recommended. This is particularly true for PT3 because of the high number of elective courses, as already mentioned.

Overall, the results indicate that the recommended courses match quite well the courses passed by students who graduated and show another way of studying to students who dropped out. The results also confirm a limitation of the proposed recommendations when the study degree program foresees many elective courses in a semester. For comparison with related work, we provide the mean F_{1GN} score for all students across programs and semesters with a value of 0.646 and the mean $Recall_{GN}$ with a value of 0.689. Depending on the semester, the scores of Ma et al. range from 0.431 to 0.472 [10] and Polyzou et al. obtain an overall mean score of 0.466 [17].

5.3 Number of Recommended Courses

We answer the questions ”a) How many courses are recommended and b) does this number differ from the number of courses passed and enrolled in by students?” in two parts.

5.3.1 Number of Recommended Courses

Figure 1 contrasts the number of recommended courses based on all neighbors and students who graduated. As already written, the recommendations are calculated with five neighbors. Their number varies between 0 and 7 in both cases. The charts show for each number the respective percentage of students grouped by status (D, G), program (AR, CM, PT), and semester (2, 3). When comparing the top and bottom charts of Figure 1, it is clear that recommendations calculated with all neighbors result in an empty set, i.e., 0 courses recommended, more frequently than recommendations calculated only with students who graduated. This confirms that the recommendations calculated only with neighbors who graduated give better results. Therefore, and as before, we consider only the recommendations calculated with neighbors who graduated in the remaining of this section.

Example CM2. In the upper half of the GN chart (bottom of Figure 2), we begin with row G-CM-2. According to the handbook, more than half of the students who graduated get six courses recommended, about 20% get five courses recommended, and the remaining students get four, three, or two courses recommended; a few students get one; no student gets an empty set. Row D-CM-2 is now under consideration. The picture looks different. About 50% of the students are recommended four or three courses, over 30% are recommended six or five courses, and the remaining students are recommended two or one course; no student is recommended an empty set.

Further investigation of the small number of courses recommended. Since some students do not receive any recommendations, see for example the rows CM3 and AR3, we examined the number of recommended courses as a function of the distance between students and their neighbors. Figure 2 shows for program CM a scatter-plot of the mean distance of the students from their neighbors (y-axis) by the number of recommended courses (x-axis) distinguishing status D and status G; semester 2 is on the left, semester 3 on the right. We can observe that when neighbors are farther

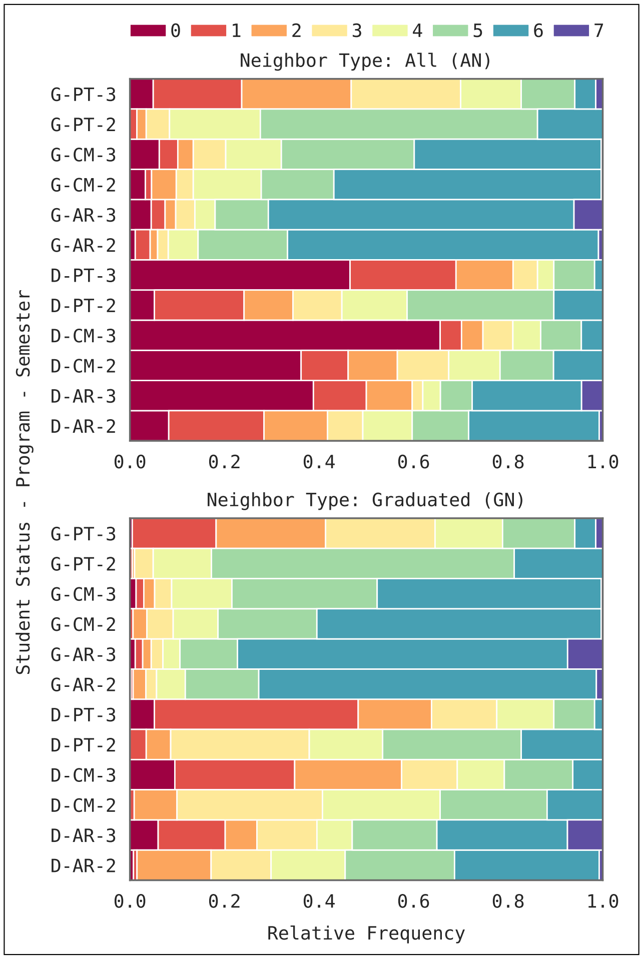

Figure 1: Distribution of students by the number of recommended courses (0 to 7), student status (D, G), program (AR, CM, PT), and semester (2, 3); top: neighbor type AN, bottom: neighbor type GN.

away, fewer courses are recommended. The trend is similar for the status dropout, though less drastic, and for the 3rd semester; it is also similar for the two other programs, not represented here. As an example, students with good grades but enrolling in only part of the courses in semesters 1 and 2, might be far from their nearest neighbors because of the imputed value of 7.0 for the courses not enrolled in.

RQ3a) Findings and Discussion. The percentage of students who receive no recommendation or only one course recommended is much smaller when the recommendations are calculated with neighbors who graduated than with all neighbors. This is especially noticeable for students who dropped out. This finding confirms again the superiority of calculating the recommendations with GN. For graduates in AR, CM, and PT in semester 2, the number of recommended courses is for the majority of the students close to the number planned in the curriculum, i.e., five or six courses. Again, PT3 differs. As is visible in the evaluation of the intersection in section 5.2, there is less agreement about the courses among the neighbors, which can be explained by a large number of elective courses in semester 3. This leads to smaller set sizes regarding course recommendations. Our results show also that students who are very different from their neighbors, especially those with status G, are likely to get few recommendations.

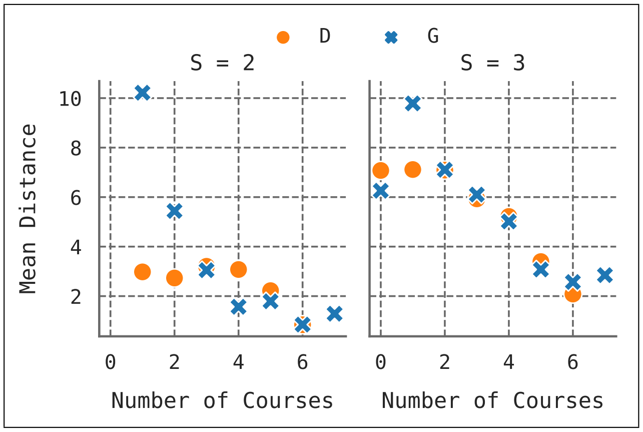

Figure 2: Mean Distance from neighbors by number of recommended courses for program CM; left: semester 2, right: semester 3. Markers and colors correspond to student status D and G.

5.3.2 *Numbers: Recommended, Enrolled, and Passed*

Table 8 provides the difference between the median number of courses recommended and the median number of courses enrolled (R - E) or passed (R - P). To better distinguish for which student groups the recommendations are closer to the actual numbers, the results are grouped by status (D, G), neighbors type (AN, GN), program, and semester PS. Note that the results with two kinds of neighbors, AN and GN, are shown for the seek of completeness. We only discuss the results calculated with neighbors who graduated, GN, as these results are better.

Example CM2. We consider first students who dropped out (D). The column (R - E) > GN has the value -1.0, which means that the number of recommended courses is on average 1 less than the number of courses the students enroll in. Comparing the number of recommended courses with the number of those passed (R - P) > GN, we see a value of 2.0, meaning that the number of recommended courses is on average 2 more than the number of courses the students pass. Considering students who graduated, we see no difference in the number of courses recommended, enrolled in, and passed on average: all values are 0.

RQ3b) Findings and discussion. On the one hand, the recommender system suggests to students who dropped out to focus on fewer courses, the column (R - E) > GN has everywhere negative values, i.e., enroll in fewer courses with the expectation that they can pass more courses instead, the column (R - P) > GN has everywhere positive values, except in PT3. On the other hand, nothing changes on average for graduates: there is no difference, except for PT3. The problem with PT3 is the lower number of recommended courses in general, as also visible in Figure 1, which can be explained by a large number of elective courses, as already written.

Table 8: Median difference between the number of courses recommended and the number of courses enrolled in (R-E) and the number of courses passed (R-P) by student status (D, G), neighbor type (AN, GN), program, and semester (PS).

	D				G			
	R - E		R - P		R - E		R - P	
PS	AN	GN	AN	GN	AN	GN	AN	GN
AR2	-2.0	-1.0	0.5	1.0	0.0	0.0	0.0	0.0
AR3	-3.0	-1.0	0.0	2.0	0.0	0.0	0.0	0.0
CM2	-3.0	-1.0	0.0	2.0	0.0	0.0	0.0	0.0
CM3	-4.0	-2.0	0.0	1.0	-1.0	0.0	0.0	0.0
PT2	-2.0	-1.0	1.0	1.0	0.0	0.0	0.0	0.0
PT3	-5.0	-4.0	-0.5	0.0	-4.0	-3.0	-3.0	-3.0

6. PRELIMINARY USER EVALUATION

The approach has been evaluated with 12 students of the study program CM as part of an assessment in the elective course “machine learning”. Students were in their 4th or 5th semester, and all performed well in their first two semesters. Beforehand, students had the possibility to hand in their records anonymously and have recommendations calculated for semesters 2 and 3. Three students made use of this possibility. The recommendations were identical to the courses that they actually passed in three cases (F_1=100%). The three other cases had an F_1 score of 90.1%, 86,1%, and 0%, respectively. The last case refers to a student with relatively good grades who enrolled in three courses only in semester 2 resulting in an average distance of 13 from the neighbors. Overall, these results confirm our assumption that, for students with good academic performance, the recommendations should closely match the courses that they pass.

The evaluation mainly consisted of a semi-guided group discussion concerning the recommendations. We report here the answers and discussion to two questions: 1. Are the recommendations understandable? 2. Would you use such a recommender system? All groups answered the first question with yes but also gave ideas for improvement. For example, they considered three to five neighbors to be the most useful, as this is the quickest and clearest way to grasp how the recommendations come about. This fits very well with the dimensions of interpretability that Guidotti et al. give [8], namely “time limitation” but also “nature of user expertise”. Six students answered the second question with yes, four with no, and two were undecided. One main reason not to use such a system was the following: seeing the grades of others can be stressful: will I perform as well as the given examples? Interestingly, an undecided student said that it might be encouraging to see that other fellow students did not always get good grades but were able to graduate. These utterances are similar to the findings in [3]. More evaluations, particularly with students who are unfamiliar with machine learning, are required to study the interpretability and related trust in the recommendations.

7. CONCLUSION AND FUTURE WORK

This paper presents a comprehensive evaluation of a novel course recommender system designed to primarily support students who face difficulties in their initial semesters and are at risk of dropping out. The evaluation utilizes data from three distinct study programs that vary in terms of their subject matter, student population, and program structure, including a program with a high number of elective courses in the third semester.

The evaluation of the first research question indicates that, overall, the recommendations lead to a reduction in the dropout rate, particularly for the targeted at-risk students who dropped out. However, the results are less conclusive for students who graduated, which may be due to the limited data available in the test set.

The evaluation of the second research question reveals that the recommended courses generally align with the courses that graduated students passed, except for the 3rd semester of program PT, which contains many elective courses. This is not the case for students who dropped out, as the recommendations suggest a different approach to their studies.

The evaluation of the third research question demonstrates that the number of recommended courses is close to the number of courses planned in the curriculum for graduating students, except for the aforementioned 3rd semester of program PT. However, for students who dropped out, the number of recommended courses is generally lower than the number of courses they enrolled in.

Overall, the evaluations have revealed two main limitations of our recommender system. Primarily, it is better suited for curricula consisting mostly of mandatory courses that all students must pass, as is often the case in the first two semesters of a program. Additionally, it recommends very few courses for students with distant neighbors, and therefore, a different approach to handling passed courses in the recommender system should be explored. However, it does allow for presenting the paths of five neighbors as an impulse.

Summing up, the paths followed by students who graduated are helpful to other students, especially those who struggle. It is worth noting that our approach to course recommendation is generalizable even if enrollment data is not stored, as is the case in some institutions. Except for comparing the number of recommended courses to the number of enrolled courses, the evaluation remains the same.

A preliminary evaluation with students indicates that the recommendations are understandable. Further research with 2nd or 3rd semester students is planned to determine how ready and willing they are to use such recommendations as well as the advantages of using sets instead of rank lists. In addition, it is necessary to evaluate whether students understand the recommendations and what additional support they need to pass all recommended courses, aside from taking fewer and different courses than they might think. As stated in the German context [7], a combination of well-orchestrated interventions usually leads to academic success.

8. REFERENCES

[1] L. Aulck, D. Nambi, N. Velagapudi, J. Blumenstock, and J. West. Mining University Registrar Records to Predict First-Year Undergraduate Attrition. In *12th International Conference on Educational Data Mining (EDM)*, pages 9–18, Montreal, Canada, 2019. International Educational Data Mining Society. https://eric.ed.gov/?id=ED599235.

[2] J. Berens, K. Schneider, S. Gortz, S. Oster, and J. Burghoff. Early Detection of Students at Risk - Predicting Student Dropouts Using Administrative Student Data from German Universities and Machine Learning Methods. *Journal of Educational Data Mining*, 11(3):1–41, 2019. https://zenodo.org/record/3594771.

[3] A. Brun, B. Gras, and A. Merceron. Building Confidence in Learning Analytics Solutions: Two Complementary Pilot Studies. In D. Ifenthaler and D. Gibson, editors, *Adoption of Data Analytics in Higher Education Learning and Teaching*, Advances in Analytics for Learning and Teaching, pages 285–303. Springer International Publishing, Cham, 2020. https://doi.org/10.1007/978-3-030-47392-1 _15.

[4] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic Minority Over-sampling Technique. *Journal of Artificial Intelligence Research*, 16:321–357, 2002. http://arxiv.org/abs/1106.1813.

[5] F. Du, C. Plaisant, N. Spring, and B. Shneiderman. Finding Similar People to Guide Life Choices: Challenge, Design, and Evaluation. In *Proceedings of the 2017 Conference on Human Factors in Computing Systems (CHI)*, pages 5498–5544, New York, NY, USA, 2017. Association for Computing Machinery. https://doi.org/10.1145/3025453.3025777.

[6] A. Elbadrawy and G. Karypis. Domain-Aware Grade Prediction and Top-n Course Recommendation. In *Proceedings of the 10th ACM Conference on Recommender Systems (RecSys)*, pages 183–190, New York, NY, USA, 2016. Association for Computing Machinery. https://doi.org/10.1145/2959100.2959133.

[7] S. Falk, M. Tretter, and T. Vrdoljak. Angebote an Hochschulen zur Steigerung des Studienerfolgs: Ziele, Adressaten und Best Practice. *IHF kompakt*, (March 2018), 2018. https://www.ihf.bayern.de/publikationen/ihf-kompakt/detail/angebote-an-hochschulen-zur-steigerung-des-studienerfolgs-ziele-adressaten-und-best-practice.

[8] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5):1–42, 2018. https://doi.org/10.1145/3236009.

[9] L. Kemper, G. Vorhoff, and B. U. Wigger. Predicting Student Dropout: A Machine Learning Approach. *European Journal of Higher Education*, 10(1):28–47, 2020. https://doi.org/10.1080/21568235.2020.1718520.

[10] B. Ma, Y. Taniguchi, and S. Konomi. Course Recommendation for University Environments. In *Proceedings of the 13th International Conference on Educational Data Mining (EDM)*, pages 460–466, Online, 2020. International Educational Data Mining Society. https://eric.ed.gov/?id=ED607802.

[11] R. Manrique, B. P. Nunes, O. Marino, M. A. Casanova, and T. Nurmikko-Fuller. An Analysis of Student Representation, Representative Features and Classification Algorithms to Predict Degree Dropout. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge (LAK)*, pages 401–410, New York, NY, USA, 2019. Association for Computing Machinery. https://doi.org/10.1145/3303772.3303800.

[12] S. Morsy and G. Karypis. Will This Course Increase or Decrease Your GPA? Towards Grade-Aware Course Recommendation. *Journal of Educational Data Mining*, 11(2):20–46, 2019. https://eric.ed.gov/?id=EJ1230292.

[13] D. Novoseltseva, K. Wagner, A. Merceron, P. Sauer, N. Jessel, and F. Sedes. Investigating the Impact of Outliers on Dropout Prediction in Higher Education. In *Proceedings of DELFI Workshops 2021*, pages 120–129, Online, 2021. Gesellschaft für Informatik e.V.z. https://nbn-resolving.org/urn:nbn:de:hbz:1393-opus4-7338.

[14] Z. A. Pardos, Z. Fan, and W. Jiang. Connectionist recommendation in the wild: on the utility and scrutability of neural networks for personalized course guidance. *User Modeling and User-Adapted Interaction*, 29(2):487–525, 2019. https://doi.org/10.1007/s11257-019-09218-7.

[15] Z. A. Pardos and W. Jiang. Designing for serendipity in a university course recommendation system. In *Proceedings of the 10th International Conference on Learning Analytics & Knowledge (LAK)*, pages 350–359, New York, NY, USA, 2020. Association for Computing Machinery. https://doi.org/10.1145/3375462.3375524.

[16] A. Polyzou and G. Karypis. Grade Prediction with Course and Student Specific Models. In *Advances in Knowledge Discovery and Data Mining. 20th Pacific-Asia Conference (PAKDD)*, pages 89–101, Auckland, New Zealand, 2016. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-31753-3 _8.

[17] A. Polyzou, A. N. Nikolakopoulos, and G. Karypis. Scholars Walk: A Markov Chain Framework for Course Recommendation. In *Proceedings of the 12th International Conference on Educational Data Mining (EDM)*, pages 396–401, Montreal, Canada, 2019. International Educational Data Mining Society. https://eric.ed.gov/?id=ED599254.

[18] M. C. Urdaneta-Ponte, A. Mendez-Zorrilla, and I. Oleagordia-Ruiz. Recommendation Systems for Education: Systematic Review. *Electronics*, 10(14):1611, 2021. https://doi.org/10.3390/electronics10141611.

[19] K. Wagner, I. Hilliger, A. Merceron, and P. Sauer. Eliciting Students' Needs and Concerns about a Novel Course Enrollment Support System. In *Companion Proceedings of the 11th International Conference on Learning Analytics & Knowledge (LAK)*, pages 294–304, Online, 2021. https://www.solaresearch.org/core/lak21-companion-

proceedings/.

[20] K. Wagner, A. Merceron, P. Sauer, and N. Pinkwart. Personalized and Explainable Course Recommendations for Students at Risk of Dropping out. In A. Mitrovic and N. Bosch, editors, *Proceedings of the 15th International Conference on Educational Data Mining (EDM)*, pages 657–661, Durham, United Kingdom, 2022. International Educational Data Mining Society. https://doi.org/10.5281/zenodo.6853008.

[21] K. Wagner, H. Volkening, S. Basyigit, A. Merceron, P. Sauer, and N. Pinkwart. Which Approach Best Predicts Dropouts in Higher Education?:. In *Proceedings of the 15th International Conference on Computer Supported Education (CSEDU)*, pages 15–26, Prague, Czech Republic, 2023. INSTICC, SciTePress. https://doi.org/10.5220/0011838100003470.