

Personalized and Explainable Course Recommendations for Students at Risk of Dropping out

Kerstin Wagner

Berliner Hochschule für Technik, Germany
kerstin.wagner@bht-berlin.de

Agathe Merceron

Berliner Hochschule für Technik, Germany
merceron@bht-berlin.de

Petra Sauer

Berliner Hochschule für Technik, Germany
sauer@bht-berlin.de

Niels Pinkwart

Humboldt-Universität zu Berlin, Germany
niels.pinkwart@hu-berlin.de

ABSTRACT

This paper presents a course recommender system designed to support students who are struggling in their first semesters of university and who are at risk of dropping out. Considering the needs expressed by our students, we recommend a set of courses that have been passed by the majority of their nearest neighbors who have successfully graduated. We describe this recommender system, which is based on the explainable k -Nearest Neighbors algorithm, and evaluate the recommendations after the 1st and the 2nd semester using historical data. The evaluation reveals that the recommendations correspond to the actual courses passed by students who graduated, whereas the recommendations and actually passed courses differ for students who dropped out. The recommendations show to struggling students a different, ambitious, but hopefully feasible way through the study program. Furthermore, a dropout prediction confirms that students are less likely to drop out when they pass the courses recommended to them.

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 in high schools differ from teaching and learning in universities. Some students cope well and keep the same level of academic performance at university as they had in high 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 focused on supporting struggling students after their 1st and 2nd semester. The final goal in developing such a system is to have it integrated in novel facilities that universities could put in place to support their diverse students better.

At the beginning of each semester in Germany, students need to decide which courses they 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 book 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 in finding answers to those questions.

In our previous work [18], most of the students mentioned that they rely on friends and acquaintances as one source of information when deciding which courses to enroll in. Further, any system assisting in enrollment should have explainable recommendations.

In this work, we propose a recommendation system to support students choosing which courses to take before the semester begins. We recommend 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 proposed in the present work does not recommend top N courses as other systems do, e.g. [10, 14]. 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 recommended set should have a lower risk of dropping out.

We describe a recommender system based on the explainable algorithm k -Nearest Neighbors (k -NN) and evaluate the recommendations given after the 1st and the 2nd semester using historical data. Although the recommendations are designed to support struggling students, every student should have access to them. By contrast, the recommendations should show a different, more academically successful way of studying to struggling students and therefore differ from the courses that they pass. More precisely, this paper tackles the following research questions:

RQ1. How large is the intersection between the set of recommended courses and the set of courses a student has passed? Are there differences between struggling and well performing students?

RQ2. Do the recommendations lower the risk of dropping out, and if so, how much?

2. RELATED WORK

Course recommendations. Supporting students in choosing courses at the start of the semester has been studied in many works. The aims of the proposed recommender systems are diverse. For example, Parameswaran et al. [13] seeks to provide students with courses that meet their constraints like availability and schedule, but also being favored and chosen by other students. Their goal is to recommend interesting courses that will also help students graduate and conduct evaluations using the academic performance data of students who have graduated. The goal of Pardos and Jiang [14] is to recommend courses “that are novel or unexpected to the student but still relevant to their interests”. The authors recommend courses based on a chosen favoured course and evaluate the results using both historical data and student feedback as part of a user study. Backenköhler et al. [1] seek to optimize recommendations by combining three aspects: a student’s preparedness for a course, the benefit of a course towards other courses, and the predicted performance in the course. The authors use historical data to evaluate how the recommendations match the student’s actual course choices. Nearer to our aim, Elbadrawy and Karypis [4] and Morsy and Karypis [10] recommend courses for which students can expect a good grade or an increase of their GPA. In both papers, the authors evaluate recommendations based on historical data, as we do.

User-centered design. Our work follows a user-centered approach as has been proposed for example in [9]. The design of our recommendations has been developed with respect to insights obtained from a semi-structured group discussion conducted with 25 student [18]. The authors of several studies have described their user-centered approach to involve stakeholders in the development of tools. De Quincey et al. [16] included students in the development of a dashboard that integrates study motivation to track engagement and predicted scores at Keele University (UK). To inform the development of a dashboard with relevant indicators to help students chose what courses to take in an upcoming academic period, Hilliger et al. [7] identified student information needs regarding course enrollment at Pontifical Catholic University (Chile) using a mixed-methods approach. Sarmiento et al. [17] described their approach of a series of co-design workshops for learning analytics tools with and for students at New York University.

Explainability. Not all machine learning algorithms used in educational data mining are explainable, such as neural networks, which are increasingly used as stated by Barredo Arritea et al. [2]. Pardos and Jiang as well as Morsy and Karypis use them in their respective works, [10, 14] for instance. According to Ning et al. [11], neighbor-based approaches are simple, justifiable, efficient, and stable. The number of neighbors, the features, and the distance function all influence the explainability of k -NN [2]. The nearest neighbors can be visualized and students understand why these recommendations are given to them. As argued by Williamson and Kizilcec in [20], “educators and learners will not trust a model that cannot easily be explained to them”. Indeed, it was a concern of our students: recommendations should be explainable [18].

Our contribution. Using student-centered design, we have developed an approach to course recommendations that is explainable to students and aims to help students who are at risk of dropping out. Our evaluation, based on historical data, distinguishes two groups: students who have dropped out and students who have graduated. First, we compare the recommendations to the set of courses that students passed; this is similar to other evaluations like [4, 12]. Second, we investigate whether our recommendations decrease the dropout-risk. This is a novelty of our work.

3. DATA AND METHODOLOGY

Data and preprocessing. For the development of the course recommendation system, data from a six-semester bachelor program of a medium-sized German university was used. The initial dataset contains 1,484 students who started their study program between winter semester 2012 and summer semester 2019. We removed three types of students: A) outliers regarding the number of passed courses based on the interquartile range (IQR) since students can receive credit for courses completed in previous study programs and thus may pass more courses in a semester than foreseen in the study handbook [12, 19], 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, and C) students without at least one record in each of the first three semesters. The final dataset contains 578 students who either graduated (status G) or dropped out (status D) and 9,500 records. The grading scale for passing a course is from best to worst [1.0, 1.3, 1.7, 2.0, 2.3, 2.7, 3.0, 3.3, 3.7, 4.0]. The grade for failing an exam is 5.0. Students must pass all mandatory courses and four elective courses to graduate. The study handbook includes a suggested courses schedule for the six semesters and students may follow this schedule or not. At any time in their studies, they are allowed to choose courses from all offered courses. It is worth noting that electives courses are scheduled in the 5th and 6th semester of the program. Table 1 shows the final number of students by student status and per semester the number of different courses, the number of academic performance records, the average number of courses passed per student, and the average grade. 134 students have the status dropout (D) and 444 the status graduate (G); in the 3rd semester (bottom line) there were 686 records concerning 20 different courses for students with the status D and 2,622 records concerning 26 courses for students with the status G. One notices that students with the status D pass fewer courses per semester and receive lower grades.

Table 1. Dataset overview by semester S (1, 2, 3) and student status (D, G). Number of courses C, Number or records R, mean number of passed courses MPC, and mean grade MG.

			C		R		MPC		MG	
D	G	S	D	G	D	G	D	G	D	G
		1	14	16	657	2,199	3.6	4.7	2.8	2.1
134	444	2	15	19	740	2,596	3.4	5.1	3.0	2.3
		3	20	26	686	2,622	3.0	5.4	3.2	2.2

Data representation. We use only data about academic performance: 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.

Missing values. To compute the nearest neighbors, we have to deal with missing values. If a student was enrolled in a course but did not take the exam, a 6.0 was imputed, if s/he was not enrolled at all, a 7.0 was imputed. This means that we rate enrolling but not taking the exam (6.0) more similar to failing (5.0) than not enrolling (7.0).

4. COURSE RECOMMENDATIONS

For our course recommendation system, we use the idea of a k -NN classifier: given a student S_x represented by the vector x_n of length n at the end of semester t , we use the majority votes of his/her already graduated neighbors to obtain a set of courses for the following semester $t+1$ that are classified as “passed” and accordingly recommended; any course not in that set is not recommended. We recommend courses for the 578 students in the dataset and then evaluate the recommendations.

Procedure. Let Y be the data of a program after semester t consisting only of graduated students S_{GR} , i.e. their vectors of grades with length n , let S_x be a student who needs a course recommendation, let C be the set of all courses of the program and let k be the number of nearest neighbors. Given C , the expected output is C_R the set of recommended courses and all courses that are in C but not in C_R are not recommended. First, we determine the k nearest neighbors of an observed student: the similarity of students is calculated using the Euclidean Distance between x and y where S_x and S_y are two students and x and y are the vectors of all their grades at the end of semester t where the length of the vectors n corresponds to the number of features that we use. The students are sorted by increasing distance and the top k vectors are selected as the neighborhood S_k of student S_x . Instead of considering each course in C , we preselect only courses that at least one of the students in S_k has passed in semester $t+1$, i.e. the grade is lower or equal to 4.0, and assign them to the set C_P of courses passed by the neighborhood. To classify each course in C_P to be recommended or not, we use the majority vote of the k neighbors: if a neighbor passed a course, it is labeled with 1 and otherwise 0. We calculate the probability P for student S_x to pass that course c as the mean of class labels, e.g. if $k=3$ and 2 out of 3 neighbors have passed the course, P is 0.66. If the mean is higher than 0.5, the course is recommended and assigned to C_R . To avoid a tie in majority voting, we use only uneven k and test our approach with $k = 1, 3, 5, 9, 19$. We check if S_x has already passed a recommended course and remove this course from the recommendation if necessary.

Baseline. We use the same approach not only with neighbors who graduated but also with all neighbors, i.e. students who dropped out and students who graduated. We expect that the recommendations differ between the two approaches 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 recommendation types with AN (all neighbors) and GN (graduated neighbors).

5. COURSES’ INTERSECTION (RQ1)

To compare the recommended courses with those actually passed, we consider the recommendations as a binary classification problem and build a confusion matrix to evaluate the recommendation as follows: a course recommended and actually passed in semester $t+1$ will be a true positive (TP), a course recommended and actually not passed will be a false positive (FP), a course not recommended but passed will be a false negative (FN), and a course not recommended and not passed will be a true negative (TN). The choice of metric is critical due to the relatively large number of courses and the resulting imbalance of recommendations or non-recommendations. Further, because we recommend a set of courses and not simply top N courses, it is crucial to measure not only that the recommendations contain all passed courses (recall) but also that they do not contain courses that students did not pass (precision). We chose the F_1 score to evaluate courses’ intersections since the F_1 score ignores true negatives, which is in our context always a high

value, and thus serves 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: $F_1 = 2 \cdot TP / (2 \cdot TP + FP + FN)$.

Results. To evaluate the quality of the recommendation, we look at the intersection and evaluate the recommendations by F_1 score. We expect a high score for students with the status G in both semesters as the recommendations for them should match closely the courses that they passed. We do not expect a high score for students with the status D; the recommendations are meant to show them another way of studying that should bring more academic success and therefore should not match closely the courses that they passed. Table 2 provides the results as mean F_1 scores of all students. To better distinguish for which student groups the recommendations are more appropriate, the results are grouped by the following factors: student status (D/G), type of neighbors (AN/GN), and semester (2, 3). In addition, we provide the difference in each row between the mean score of the baseline AN and the mean score of GN. The differences in the number of neighbors k are not shown because no large differences emerged when k varied.

Findings. There is a difference between students who struggle and students who perform well. More precisely: A) AN-based and GN-based recommendations: The mean F_1 score for dropouts and graduates is higher for GN than for AN. B) Semester: The mean F_1 score is higher after the 1st for the 2nd semester than after the 2nd for 3rd semester. C) Graduates and dropouts: The mean F_1 score is higher for graduates than for dropouts. The mean F_1 score is high for graduates: this confirms our expectation that recommendations closely match the courses passed by graduates. For students with status D, the mean F_1 score tends to be low: the recommendations show another way of studying.

Table 2. Courses’ intersection based on mean F1 score of all students for semester S and recommendation types AN and GN as well as their differences (Diff) by student status (D, G).

S	D			G		
	AN	GN	Diff	AN	GN	Diff
2	0.471	0.509	0.038	0.843	0.861	0.017
3	0.282	0.302	0.020	0.809	0.836	0.027

6. CHANGES IN DROPOUT RISK (RQ2)

A dropout prediction was performed using the following two steps: 1) based on the actual enrollment and exam information, a model has been trained to predict the two classes dropout or graduate; 2) the model of step 1 is used again to predict dropout or graduate after replacing the actual enrollment and exam information by the calculated recommendations. We call dropout risk the proportion of students who are predicted “dropout” by a model. To determine whether or not the recommendation approach helps to lower the dropout risk, we analyze the difference of these two dropout predictions.

Feature set. As investigated in [8], 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, and the individual courses are relevant accordingly, we use the courses taken by students as features and the grades obtained by students as their values.

Model training. To detect a change in the dropout risk in step 2, the models should be as accurate as possible which we aimed to achieve through two approaches: A) train different types of algorithms including hyperparameter optimization (Logistic

Regression, LASSO Regression, Decision Tree, k -Nearest Neighbors, Support Vector Classifier with different kernels (radial basis function, linear, polynomial), Random Forest), and B) use algorithm-independent parameters for optimization since we realized that hyperparameter optimization alone was insufficient. B1) Feature selection: we removed features with a low number of actual grades and tried different thresholds as a minimum number of actual grades for a course: (1, 5, 9, 19). B2) Training data balancing: we used two common techniques: Synthetic Minority Over-sampling Technique (SMOTE) [3] and RandomOverSampler (ROS). Both implementations are from the Python library imbalanced-learn. B3) Decision threshold moving: 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 the values: 0.1, 0.3, 0.7, 0.9.

Models selection. To emphasize that both correct dropouts and correct graduates are important for dropout risk prediction, we evaluated our models based on the test data with *Balanced Accuracy* (BACC) as the mean of the recall for class 1 (dropout) and recall for class 0 (graduate). The data sets sorted by the start of study were split into 80% training data and 20% test data, so that risk prediction is done for students who started their studies last. We trained models for both semesters $t=2$ and $t=3$ with actual grades and used the best models to evaluate a change in dropout risk in step 2. In both cases, RF achieved the highest BACC for step 1 (2nd semester: 0.859, 3rd semester: 0.935). The algorithm-independent parameters belonging to the models are: features selection (2nd: 0, i.e. no features were removed, 3rd: 1), decision threshold (2nd: 0.3, 3rd: 0.5), and balancing the training data (2nd: SMOTE, 3rd: ROS). Regarding hyperparameter optimization, we did not observe any improvements in BACC, which we relate to the small size of the training sets.

Step 2 dropout prediction. We used the selected models to predict dropout based on the recommendations. Since we assume the student will pass the recommended courses, we need a grade between 1.0 and 4.0 for step 2. If we had a grade in the records for that student and a recommended course, we used it. If the student had dropped the course or failed it, we imputed the average of two medians: the median of all the grades that the student has earned so far and the median of the historical grades for that course. We evaluated this imputation with the data we use in this work and obtained an *Root Mean Square Error* (RMSE) of 0.634, which is comparable with the RMSE from 0.63 to 0.73 in [4, 15].

Results. Table 3 shows three proportions of students who are predicted as dropout, P1, P2_AN and P2_GN and the differences between these proportions. P1 is the prediction based on actual enrollment and exam data. P2 AN corresponds to step 2; the dropout prediction uses the courses recommendations calculated with all neighbors while P2_GN uses the recommendations calculated with graduate neighbors. The differences in the number of neighbors k are not considered because no large differences emerged when k varied: P2_AN and P2_GN are the average values from the risk predictions, based on the test data set. The three columns on the right provide the differences in the predictions: P2_AN vs P1, P2_GN vs P1, P2_GN vs P2_AN. For example, 81.4% of the actual dropout students for semester 2 are predicted as dropout in the first prediction, 75.8% using AN-based recommendations, and 70.2% using GN-based recommendations. The dropout risk based on the GN recommendation is 5.6% lower than the prediction based on the AN recommendation.

Findings. It turns out that, with our strong assumption that students will pass the recommended courses, the risk of dropping out can be

reduced. More precisely: The proportion of students predicted as dropout are lower when the predictions are AN- and GN-based in step 2 than using the actual enrollment and exam data in step 1. The GN-based recommendations provide a lower dropout risk compared to the AN-based recommendations. The proportion of students predicted as dropout are lower for the 3rd than for the 2nd semester using AN- and GN-based recommendation. The dropout risk reduction in step 2 is higher for dropouts than for graduates.

Table 3. Mean predicted risk in step 2 dropout prediction (P2) for recommendation types AN and GN compared to step 1 (P1) by student status ST (D, G) and semester (S).

ST	S	P1	P2_AN	P2_GN	P2_AN vs P1	P2_GN vs P1	P2_GN vs P2_AN
D	2	0.814	0.758	0.702	-0.056	-0.112	-0.056
	3	0.884	0.665	0.293	-0.219	-0.591	-0.372
G	2	0.096	0.074	0.060	-0.022	-0.036	-0.014
	3	0.014	0.047	0.005	0.033	-0.008	-0.041

7. DISCUSSION AND CONCLUSION

In this paper, we have presented an explainable course recommender system designed primarily to support students who are struggling after their 1st or 2nd semester at university. The recommendations are based on the explainable k -NN algorithm and are built by selecting the courses that most of the nearest neighbors who graduated have passed. We have evaluated our approach on historical data in two ways. First, we have compared the recommendations with the set of the courses that students have passed using the F_1 score. Second, we have investigated whether students are less likely of dropping out when following the recommendations. Further, we also evaluate the impact of choosing nearest neighbors from the set of students who dropped out and graduated, our baseline, instead of choosing them only from the set of students who graduated.

The F_1 score evaluating the recommended courses is higher when the neighbors are chosen from the set of students who graduated, as can be seen in Table 2. It is particularly high, mainly over 80%, for students with the status graduate, which confirms that, for them, the recommendations match closely the courses that they pass. Consistent with this finding, the number of students who are predicted with the status dropout is smaller when the recommendations are used in the prediction rather than the actual data. Preliminary work shows that these findings generalize to other degree programs.

The results suggest that the provided recommendations would help more students to graduate if the recommendations are both ambitious and realistic: students indeed do pass the courses recommended to them. A closer look at the recommendations reveals that a small number of students receive an empty set, which should be examined in detail. Further, it still needs evaluations with students in their 1st or 2nd semester on how ready and willing they are to use such recommendations, and which extra support they need to pass all courses recommended to them. As stated in [6] in the German context, most of the time a combination of well-orchestrated interventions brings academic success.

In terms of k -NN, it would be worth testing the possibilities of multilabel learning as presented in [21] and whether the same approach could be used for planning over several semesters, as proposed in [5]. Finally, we would like to investigate which other explainable approaches are equally visualizable and understandable to students.

8. REFERENCES

- [1] Backenköhler, M., Scherzinger, F., Singla, A., and Wolf, V. 2018. Data-Driven Approach towards a Personalized Curriculum. In *Proceedings of the 11th International Conference on Educational Data Mining*, 246–251.
- [2] Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Benetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion* 58, 82–115.
- [3] Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. 2002. SMOTE: Synthetic Minority Over-sampling Technique. *jair* 16, 321–357.
- [4] Elbadrawy, A. and Karypis, G. 2016. Domain-Aware Grade Prediction and Top-n Course Recommendation. In *Proceedings of the 10th ACM RecSys Conference on Recommender Systems*. Association for Computing Machinery, 183–190. DOI=10.1145/2959100.2959133.
- [5] Erzhuo Shao, Shiyuan Guo, and Zachary A. Pardos. 2021. Degree Planning with PLAN-BERT: Multi-Semester Recommendation Using Future Courses of Interest. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence. AAAI Special Track on AI for Social Impact*. 35, 17, 14920–14929.
- [6] Falk, S., Tretter, M., and Vrdoljak, T. 2018. Angebote an Hochschulen zur Steigerung des Studienerfolgs: Ziele, Adressaten und Best Practice. *IHF kompakt*, March 2018.
- [7] Hilliger, I., Laet, T. de, Henríquez, V., Guerra, J., Ortiz-Rojas, M., Zuñiga, M. Á., Baier, J., and Pérez-Sanagustín, M. 2020. For Learners, with Learners: Identifying Indicators for an Academic Advising Dashboard for Students. In *Proceedings of the 15th European Conference on Technology Enhanced Learning. Addressing Global Challenges and Quality Education*. Lecture Notes in Computer Science. Springer International Publishing, Cham, 117–130. DOI=10.1007/978-3-030-57717-9_9.
- [8] Manrique, R., Nunes, B. P., Marino, O., Casanova, M. A., and Nurmikko-Fuller, T. 2019. 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*. Association for Computing Machinery, New York, NY, USA, 401–410. DOI=10.1145/3303772.3303800.
- [9] Martinez-Maldonado, R., Pardo, A., Mirriahi, N., Yacef, K., Kay, J., and Clayphan, A. 2016. LATUX: an Iterative Workflow for Designing, Validating and Deploying Learning Analytics Visualisations. *Learning Analytics* 2, 3, 9–39.
- [10] Morsy, S. and Karypis, G. 2019. Will this Course Increase or Decrease Your GPA? Towards Grade-aware Course Recommendation. *Journal of Educational Data Mining* 11, 2, 20–46.
- [11] Ning, X., Desrosiers, C., and Karypis, G. 2015. A Comprehensive Survey of Neighborhood-Based Recommendation Methods. In *Recommender Systems Handbook*, F. Ricci, L. Rokach and B. Shapira, Eds. Springer US, Boston, MA, 37–76. DOI=10.1007/978-1-4899-7637-6_2.
- [12] Novoseltseva, D., Wagner, K., Merceron, A., Sauer, P., Jesel, N., and Sedes, F. 2021. Investigating the Impact of Outliers on Dropout Prediction in Higher Education. In *Proceedings of DELFI Workshops 2021*. DELFI 2021 - 19. Fachtagung Bildungstechnologien der GI. Hochschule Ruhr West, 120–129.
- [13] Parameswaran, A., Venetis, P., and Garcia-Molina, H. 2011. Recommendation systems with complex constraints. *ACM Transactions on Information Systems* 29, 4, 1–33.
- [14] Pardos, Z. A. and Jiang, W. 2020. Designing for serendipity in a university course recommendation system. In *Proceedings of the 10th International Conference on Learning Analytics & Knowledge*. Association for Computing Machinery, New York, NY, USA, 350–359. DOI=10.1145/3375462.3375524.
- [15] Polyzou, A. and Karypis, G. 2016. Grade Prediction with Course and Student Specific Models. In *Proceedings of the 20th Pacific Asia Conference on Knowledge Discovery and Data Mining*, 89–101. DOI=10.1007/978-3-319-31753-3_8.
- [16] Quincey, E. de, Briggs, C., Kyriacou, T., and Waller, R. 2019. Student Centred Design of a Learning Analytics System. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*. Association for Computing Machinery, New York, NY, USA, 353–362. DOI=10.1145/3303772.3303793.
- [17] Sarmiento, J., Campos, F., and Wise, A. 2020. Engaging Students as Co-Designers of Learning Analytics. In *Companion Proceedings 10th International Conference on Learning Analytics & Knowledge*, 29–32.
- [18] Wagner, K., Hilliger, I., Merceron, A., and Sauer, P. 2021. Eliciting Students’ Needs and Concerns about a Novel Course Enrollment Support System. In *Companion Proceedings of the 11th International Conference on Learning Analytics & Knowledge*, Online, 294–304.
- [19] Wagner, K., Merceron, A., and Sauer, P. 2020. Erste Untersuchungen zur Notenprognose für ein Kursempfehlungssystem. In *Proceedings of DELFI Workshops 2020*, 102–113. DOI=10.18420/delfi2020-ws-112.
- [20] Williamson, K. and Kizilcec, R. F. 2021. Effects of Algorithmic Transparency in Bayesian Knowledge Tracing on Trust and Perceived Accuracy. In *Proceedings of the 14th International Conference on Educational Data Mining*, 338–344.
- [21] Zhang, M.-L. and Zhou, Z.-H. 2007. ML-KNN: A lazy learning approach to multi-label learning. *Pattern Recognition* 40, 7, 2038–2048.