Estimate Undergraduate Student Enrollment in Courses by Re-purposing Recommendation Tools

Md Akib Zabed Khan
Florida International University
mkhan149@fiu.edu

Agoritsa Polyzou
Florida International University
apolyzou@fiu.edu

Abstract

Resource allocation in educational institutions is a very challenging task in higher education. To prepare for every new semester, academic administration faces various challenges in allocating instructors, classrooms, sessions, teaching assistants, and laboratories for different possible courses considering students' needs and the limited available resources. Predicting the number of students enrolled in a specific class in the next semester can help with this task. To address this problem, we investigate various machine learning models (direct and indirect methods) using different features of course enrollment data of past students to predict the number of enrollments in possible courses in the upcoming semester. In this work, we propose to use a course recommendation model as a first step to generate suggestions for students, and then, use those to estimate student enrollment in the courses of the next semester. We test four course recommendation models, two time series models, three regression models, and three baseline approaches for course enrollment prediction. The experimental evaluation demonstrates that our proposed approach achieves good behavior and similar or better performance compared to other competing approaches to predict student enrollment in courses.

Introduction

In higher education, academic administration needs to make many decisions ahead of time, well before the start of a new semester. Preparing the course offerings for every semester is a very difficult task because administrators need to balance students' interests and the limited resources they have. They must make a plan to allocate available instructors, classrooms, sessions, and laboratories for each offered course for the next semester based on their experiences working at the department. *Student enrollment prediction* in a course in the upcoming semester is very important to accomplish the mission and goals of a department (Hopkins 1981). Interactive models focus on how information systems can aid and enhance users in making decisions more efficiently and effectively (Cecez-Kecmanovic 2002). Artificial Intelligence (AI)-based predictive models can be very useful in making necessary decisions for course offerings, resource (classrooms, laboratories) allocation of a department, and developing new strategies for the betterment of students' pathways to graduation (Ward 2007); (Huarng and Hui-Kuang Yu 2013); (Kaur, Polyzou, and Karypis 2019).

In the literature, there are numerous predictive models to estimate the number of students enrolled in a department of a university during the admission process of freshmen students (Ujkani, Minkovska, and Stoyanova 2021); (Slim et al. 2018); (Saini and Jain 2013); (Davidson 2005). As the number of enrollments of students changes over time, many researchers develop different time series models to analyze this type of data (Lee, Efendi, and Ismail 2009); (Ismail and Efendi 2011); (Chen, Li, and Hagedorn 2019). Other researchers estimate how many students will be admitted to college by using different types of data, e.g., enrollments in previous years, student and college characteristics, demographic information of students, questionnaire surveys, etc (Nandeshwar and Chaudhari 2009); (Soltys et al. 2021). There are some predictive models that estimate students' enrollment in the upcoming semester (Watkins and Kaplan 2018); (Egbo and Bartholomew 2018); (Shao et al. 2022).

To address the issue of student enrollment prediction in courses in the upcoming semester, we focus on the particular student population that is about to enroll in the upcoming semester. We propose a two-phase approach; build a course recommendation system for these students first, and then use the courses suggested by the model to estimate student enrollment in upcoming courses. We explore different machine learning-based predictive models, time series forecasting models, and other indirect approaches by utilizing other tools (course recommendation systems, and a classification model). Our proposed **prediction from recommendation (PfR)** models use course recommendations to make predictions about student enrollment in courses. Concisely, we calculate the number of times we recommend a course for all the students which serves as our prediction of the number of students that will enroll in that course in the upcoming semester. Our contributions are as follows:

- We propose to use course recommendation models to predict the number of students enrolled in the courses that will be offered in the next semester.
- We examine how four recent and commonly used course recommenders perform when used within our PfR model.

- We compare our proposed approach against different types of approaches that have been considered for this problem (random forest regressor, support vector machine regressor, Gaussian process regression, logistic regression) or other relevant problems (time series models), using a real-world dataset from a big public institution.

Our proposed approach performs better than the alternative approaches for the majority of the courses. As a result, it can be a useful tool for departmental administration, supporting decision-making and preparing ahead to allocate resources for the upcoming semesters. In turn, this will result in better student satisfaction in higher education.

Related Work

Administration Support

Future student enrollment prediction in courses is a nontrivial task. It involves past, new, and transfer students, and their evolving interests over time. The increasing volume of students puts an additional strain on the administration that plans future course offerings. As a result, waiting lists get full relatively quickly, and students may fail to enroll in courses that they want or even need (i.e., required courses) in order to graduate. This is an understudied topic but it can have a significant impact on students' planning and progress.

To provide administration support, some researchers develop different machine learning models to predict students' enrollment using the course registration history of past students, historical course enrollment information about offered courses (without students' course enrollment information), and course evaluation data. (Watkins and Kaplan 2018) propose a design of student enrollment prediction tool by applying time series models where the Gaussian Processes model works better than linear regression, multilayer perceptron, and support vector machines (SVM) for regression models and other time series models, autoregressive integrated moving average (ARIMA) and exponential smoothing (ETS). (Lee 2020) and (Shao et al. 2022) present several enrollment prediction models (SVM for regression, logistic regression, Markov model, random forest regressor), and among them, the random forest regressor model outperforms others. (Aksenova, Zhang, and Lu 2006) present a student enrollment prediction model using SVM for regression and then utilize a tool to generate a rule-based predictive model using the initial SVM predictions. (Egbo and Bartholomew 2018) present a multi-layer feed-forward neural network model to predict student enrollment. Another multi-layer perceptron (MLP) model has been proposed for courses in an e-learning platform (Kardan et al. 2013).

(Wang et al. 2014) introduce a fuzzy time series forecasting model to predict students' enrollment. They use the yearly difference of enrollment as the main domain to develop the fuzzy system. (Lavilles and Arcilla 2012) also use three time series forecasting models (simple moving average, single exponential smoothing, double exponential smoothing) to predict the number of student enrollment and incorporate the double exponential smoothing model (best one for their data) to their school management system. (Biswas et al. 2023) propose an algorithmic approach to solve course enrollment challenges promoting fairness.

Using regression models and time series forecasting for predicting student enrollment has limitations. These models primarily recognize patterns and trends from past semesters or years, operating at a higher, more generalized level. They do not delve into detailed information about the individual enrolled students. To address this limitation, we incorporate information about the current student body, analyze interactions between students and courses, and initially develop a Course Recommendation System (CRS). Subsequently, we leverage its course recommendations to enhance the precision of predicting students' enrollment in courses.

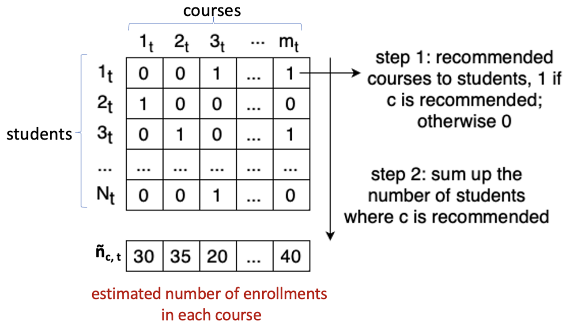
Other Applications of Time Series Models

Time series analysis is widely applied for forecasting stock prices for a long time (Kraft and Kraft 1977); (Mondal, Shit, and Goswami 2014); (Mehtab and Sen 2020). Two deep learning-based regression models and an integrated artificial neural network model trained with meta-heuristic algorithms have been proposed to analyze time-series data and estimate stock prices (Mehtab and Sen 2020); (Shahvaroughi Farahani and Razavi Hajiagha 2021). Besides, a fuzzy time series model integrating granular computing has also been proposed to predict stock prices (Chen and Chen 2015). Time series models are also used to solve different problems in climate modeling, medicine, biological sciences, finance and e-commerce, electricity, and educational data mining (EDM) fields (Lim and Zohren 2021); (Deb et al. 2017); (Fakhrazari and Vakilzadian 2017). Within EDM, researchers used time series analysis to predict students' performance in upcoming courses, the number of students' enrollments, and for behavior-based dropout prediction (Chen and Cui 2020); (Wang et al. 2014); (Haiyang et al. 2018).

Course Recommendation Models

Next, we examine the state of the art for the problem of course recommendation. Many researchers analyzed real-world course enrollment and course description datasets gathered from their respective universities and colleges (Al-Badarenah and Alsakran 2016); (Pardos and Jiang 2020); (Wong 2018). Different machine learning methods have been used to build CRS (Bendakir and Aïmeur 2006). (Pardos, Fan, and Jiang 2019) propose a combination of long short-term memory (LSTM) networks and skip-gram model to recommend courses considering the preferences of students and a course2vec model to recommend serendipitous courses for the next semester (Pardos and Jiang 2020). An RNN model and a knowledge graph recommend courses considering the career goals of students in (Jiang, Pardos, and Wei 2019); (Nguyen, Vu, and Ly 2022).

(Polyzou, Nikolakopoulos, and Karypis 2019) propose a random-walk-based approach, Scholars Walk, capturing the sequential transitions of courses semester-by-semester. Besides, a PLAN-BERT model has been proposed to recommend multiple consecutive semesters in (Shao, Guo, and Pardos 2021). CourseBEACON and CourseDREAM have also been proposed using deep learning to recommend courses considering the relationship among courses taken



courses

students	1_t	2_t	3_t	...	m_t
1_t	0	0	1	...	1
2_t	1	0	0	...	0
3_t	0	1	0	...	1
...	...	...	...	...	...
N_t	0	0	1	...	0
$\tilde{n}_{c,t}$	30	35	20	...	40

step 1: recommended courses to students, 1 if c is recommended; otherwise 0

step 2: sum up the number of students where c is recommended

estimated number of enrollments in each course

Figure 1: *Our prediction from recommendation (PfR) approach for course enrollment prediction.*

within a semester and the sequential transitions of courses over semesters (Khan and Polyzou 2023).

Proposed Approach

We consider that in the t-th semester, we have N_t students taking courses. Each student, i, takes a set of courses, $\mathcal{C}_{i,t}$, in the t-th semester. Let $n_{c,t}$ be the number of students enrolled in course c in the t-th semester, i.e.,

$$n_{c,t} = \sum_{i=1}^{N_t} \mathbb{I}(c \in \mathcal{C}_{i,t}), \tag{1}$$

where $\mathbb{I}()$ is the indicator function that returns 1 if the statement is true, else 0. Instead of directly trying to estimate $n_{c,t}$, we propose to use a two-step approach. First, using a course recommendation system, we generate a recommended set of courses, $\tilde{\mathcal{C}}_{i,t}$ for all the students who will take courses in the upcoming (target) semester, t. More specifically, we recommend to each student the number of courses that they want to register for. As a result, the total number of recommendations is equal to the total number of student enrolments. We assume that students will actually follow the recommendations, and register for these courses. Then, we sum up the number of students for whom we recommended a specific course and this number is the estimated number of enrollments for that course in the next semester.

Let $\mathcal{R}_i$ be the set of courses we recommend for student i for a semester. O_c denotes the list of offered semesters. We need to estimate $n_{c,t}$ before students register for courses early at the $(t-1)$-th semester, i.e., without knowing the actual course selection of the students, $\mathcal{C}_{i,t}$. By using the recommended courses $\mathcal{R}_i$ for all students, we have estimated $\tilde{\mathcal{C}}_{i,t}$ which we can in turn use to estimate the student enrollment in the courses offered in the t-th semester. In Eq. 1, we will replace $\mathcal{C}_{i,t}$ with $\tilde{\mathcal{C}}_{i,t}$, to get the estimate of $n_{c,t}$, $\tilde{n}_{c,t}$. Our proposed approach is illustrated in Figure. 1 where N_t is the number of students and m_t is the number of available courses to be offered in t-th semester.

While our approach is simple, it does account for two important aspects of the course enrollment prediction. First, it considers the number of students enrolled in the department. Over the years, the number of students in a program may increase or decrease, so our model will be able to capture these trends. Additionally, our estimation is based on personalized recommendations for our students. This means that we capture their possible future paths, and make sure that we reserve a seat for the courses that are the most likely to take considering their registration history.

Experimentation evaluation

Dataset

We have used a real-world dataset collected from Florida International University (FIU), a public US university, from summer 2014 until spring 2022. Our dataset includes the course registration history of undergraduate students in the Computer Science (CS) department. We do not use any demographic data for the students. The dataset is anonymized and we have received IRB approval for its use. Initially, there are $1,189$ unique courses and 133 CS departmental unique courses of $3,703$ students in our dataset. We only take into account data from students who earned a degree successfully. We keep only instances with letter grades A, A-, B+, B, B-, C+, C, and P (pass); grades less than C typically do not apply towards degree requirements (Morsy and Karypis 2019). We also remove courses that appear fewer than ten times and courses that are offered in less than four different semesters. Then we split the data into train, validation, and test sets. For testing purposes, we utilize the last three semesters: summer 2021, fall 2021, and spring 2022. The preceding three semesters, summer 2020 to spring 2021, are used for validation and model selection. The training set contains the remaining course registration history before summer 2020, spanning almost seven years.

In the validation and test sets, we exclude courses that are not present in the training set. Additionally, we eliminate the students that have less than three semesters. After preprocessing, we have the course registration history of 3,324 students with 334 unique courses in total, out of which, 66 are CS departmental courses. There may be several instances of each student, one for each semester that can be considered as the target semester. There are 2,968, 1,228, and 655 students in the training, validation, and test sets, respectively. The corresponding number of target semesters is 13,990, 2,729, and 1,251, respectively. While we cannot publicly share our student data, the code for our methods can be found here: https://github.com/PolyDataLab/CourseEnrollmentPred.

Implemented approaches

We implement three baselines that only use the number of prior course enrollments without building any machine learning model. We explore four direct approaches implementing popular machine learning models and time series models. We test five indirect approaches (one based on classification and four based on CRS proposed in this paper). The methodologies considered are illustrated in Fig. 2.

Indirect Method: Prediction from Recommendation

1. PfR(LSTM) We re-implement the LSTM-based CRS similar to (Pardos, Fan, and Jiang 2019). We create a multi-hot representation of courses per semester for each student and feed the sequence of representations of semesters of

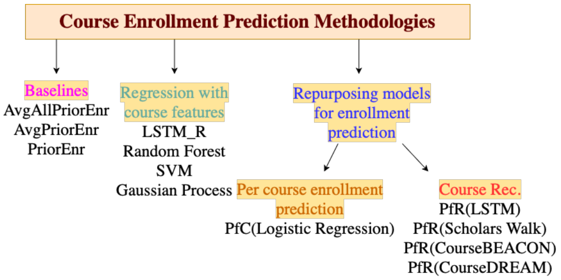


Figure 2: *Methods for student enrollment prediction.*

each student as input to the LSTM networks architecture to predict the courses in the last semester. We recommend the courses with the highest probability values.

2. PfR(Scholars Walk) We create a Markov chain by using course transitions from semester to semester (Polyzou, Nikolakopoulos, and Karypis 2019). Considering the courses that the student took in the prior semester as the starting point, we perform a random walk with restarts on this course-to-course graph.

3. PfR(CBEACON) For the CourseBEACON model, we create a correlation matrix, where we calculate the frequency of a pair of courses taken together within a semester, and then normalize the matrix (Khan and Polyzou 2023). We feed the sequence of representations of semesters' courses and the correlation matrix as input to the LSTM network architecture. We take the hidden signal from the last LSTM layer and use a correlation-sensitive score predictor to estimate scores for all the available courses to be recommended.

4. PfR(CDREAM). In the CourseDREAM model, we use average pooling to build a representation of the set of courses taken within a semester (Khan and Polyzou 2023). We pass the sequence of representations of semesters for each student to an LSTM architecture. From the last hidden signal, we get the probability of each course being recommended to the target student for the upcoming semester.

Parameter search space In the LSTM model, we have tried hidden layers: [1,2,3], embedding dimensions: [16,32,64], hidden units: [32,64,128,256] in an LSTM layer, and dropout rates: [0.3,0.4]. For the Scholars Walk model, we have tried the number of steps allowed: [1,2,3,4,5]; $alpha$: [1e-4, 1e-3, 1e-2, 1e-1, 0.2, 0.4, 0.6, 0.7, 0.8, 0.85, 0.9, 0.99, 0.999]; and $beta$ values from 0 to 1.6 with step 0.1. For the CourseBEACON, we have tried α: [0.1,0.3,0.5,0.7,0.9] which balances the importance of intra-semester dependencies, and sequential transition of courses over semesters. In the LSTM architecture of the CourseBEACON, we have tried the hidden layers: [1,2,3], embedding dimensions [16,32,64], hidden units: [32,64,128] of an LSTM layer, and dropout rates: [0.3,0.4]. For the CourseDREAM, in the LSTM networks, we have tried the number of hidden layers: [1,2,3], embedding dimensions: [8,16,32], and dropout rates: [0.3,0.4,0.5,0.6].

Indirect Method: Predict from Classification, PfC(LR) We can use classification (with logistic regression, LR), **PfC(LR)**, to predict course enrollment as described in (Lee 2020). Instead of directly estimating $\tilde{n}_{c,t}$, we use the students' course enrollment history to build a classification model for each course c that predicts if a student will enroll for course c. We build as many logistic regression models as the number of available courses in our dataset. For each course, we count how many students we predict that will take course c, i.e., $c \in \mathcal{C}_{c,t}$, and that is the final student enrollment prediction for course c. Using the training data, for each course taken in the prior 6 semesters (i.e., for the last two years), we take all the prior courses of a student (who took courses in the last two years) and make a binary vector of size m for each student, where m is the number of all available courses. We set 1 for the courses taken by that student; otherwise, 0. The target label is set to 1 if the specific course (in the target semester) is taken by that student; otherwise, we set it to 0. We train a model with the data for each course implementing a logistic regression algorithm using the scikit-learn library (Pedregosa et al. 2011).

Direct Method: Baseline Approaches These approaches use only the registration history of courses.

1. AvgAllPriorEnr We take the average of all prior enrollments for each course. The equation is: $\tilde{n}_{c,t} = \sum_{j \in O_c} n_{c,j} / |O_c|$, where $n_{c,j}$ is the enrollment of course c at semester j, O_c is the list of offered semesters and $|O_c|$ is the number of all prior semesters when course c was offered.

2. AvgPriorEnr Considering that students' preferences change over time, we calculate the average enrollment only over the last four prior semesters for a course.

3. PriorEnr In this approach, the estimated number of enrollment in a course is the exact number of enrollment in that course in the last semester it was offered. The assumption is that the number of students in a course will be similar to the most recent offering of the course.

Direct Regression Models We build two time series models, LSTM for regression (LSTM_R) and Gaussian process regression (GPR), and two regression models, random forest (RF) and support vector machine (SVM), to predict the number of enrollments in courses directly. For building the LSTM_R model, from the training data, we take the enrollments of each course in 9, 12, or 15 prior terms and use only these numbers to predict the enrollments in that course in the next semester. For the other models, using the training data for each course, we extract ten features that describe it, i.e., student enrolments in this specific course in the last 4 semesters (4 attributes), level of the course (5 asymmetric binary attributes), number of average student enrollment in all courses in the prior semester. For all the models (including LSTM_R), if a course is not offered in any of the prior semesters, we take the average number of enrollment in all courses in that semester as a feature (Lee 2020).

1. LSTM_R We implement a time series model building neural networks to predict course enrollment (Chniti, Bakir, and Zaher 2017). As we have three terms in a calendar year, we consider a window of 3 semesters for each LSTM unit. We pass the input sequences to the LSTM networks with 128 hidden units and then add a dense layer of 64 hidden units as shown in Figure 3.

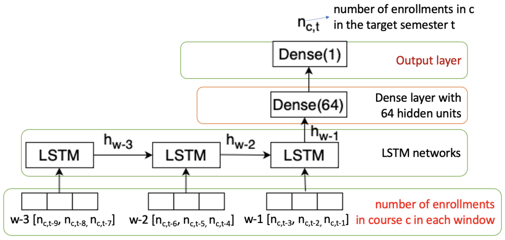

$n_{c,t}$
number of enrollments in c
in the target semester t
Dense(1)
Output layer
Dense(64)
Dense layer with
64 hidden units
h_{w-1}
LSTM
h_{w-3}
LSTM
h_{w-2}
LSTM
LSTM networks
w-3 [$n_{c,t-9}$, $n_{c,t-8}$, $n_{c,t-7}$]
w-2 [$n_{c,t-6}$, $n_{c,t-5}$, $n_{c,t-4}$]
w-1 [$n_{c,t-3}$, $n_{c,t-2}$, $n_{c,t-1}$]
number of enrollments
in course c in each window

Figure 3: *The architecture of LSTM_R model*

2. Random Forest As in prior work (Lee 2020); (Watkins and Kaplan 2018); (Shao et al. 2022), we implement a random forest regressor model using the scikit-learn library (Pedregosa et al. 2011). We use ten features for each course as described above. The target value is the number of students enrolled in that course in the target semester.

3. Support Vector Machine (SVM) We implement a SVM for regression model (Lee 2020); (Watkins and Kaplan 2018). We use those ten features for each course as described above to train our SVM model. We build the model utilizing linear kernel SVM for regression algorithm available in the scikit-learn library (Pedregosa et al. 2011).

4. Gaussian process regression (GPR) We build a GPR model using the same features used to build the SVM model (Watkins and Kaplan 2018). In this model, the target function is represented as a Gaussian distribution, and a Gaussian random value is assigned to each point in the input. We train the model using a kernel consisting of WhiteKernel and DotProduct utilizing the scikit-learn library (Pedregosa et al. 2011). For evaluation, we calculate the mean value as the prediction for the number of enrollments. As described in (Watkins and Kaplan 2018), we also considered lag time = 3 to train the Gaussian process as we have 3 terms in each academic year, but the performance was not improved.

Parameter search space: In the random forest regressor, we use max_depth $= 2$ for each decision tree. In the linear kernel SVM for regression model (SVR), we have used the regularization parameter, C $= 1$, and epsilon $= 2$ which is the margin of the tolerance for error. In the Logistic Regression model, we do not have any parameters to tune.

Evaluation Metrics The metrics we will use to evaluate course recommendation (our first step) are **Recall**@k = # of correct recommendations/# of courses in target semester, and **Precision**@k = # of correct recommendations/# of courses recommended. We compute these metrics @k, i.e., we recommend to each student as many courses as the student plans to take in the target semester, k. In this case and when we consider all the courses, Recall@k is the same as Precision@k, so we only show recall. However, our dataset includes many courses from different departments. Given that our ultimate goal is to support administrative decision-making in the CS department, we separately evaluate how the models perform only for the courses offered by the CS department. We use the number of relevant (correct) CS recommendations divided by the number of CS courses in the target semester (**Recall**@CS) or the number of CS courses that we recommended (**Precision**@CS).

Model	Recall@k	Recall@CS	Prec@CS
LSTM	0.269	0.314	0.335
Scholars Walk	0.285	0.367	0.359
CBEACON	0.301	0.393	0.362
CDREAM	**0.320**	**0.422**	**0.388**

Table 1: Recommendation performance for Recall@k (all courses), Recall@CS, and Precision@CS (only CS courses).

To evaluate student enrollment prediction in the target semester, we compute the error of the predicted compared to the actual enrollment, i.e., error$_{c,t} = |\tilde{n}_{c,t} - n_{c,t}|$. To aggregate the errors over all the courses, we calculate mean absolute error (MAE), and the MAE separately over the courses for which we estimate more (MAE+) or fewer students (MAE-) than actually enrolled. For example, if 30 students are actually enrolled in a course and our prediction is 28, we underestimate the enrollment (under-prediction error = 2) and include this error in the MAE- calculation.

Results

Course Recommendation

We present the recommendation results for all courses and only CS departmental courses in Table 1. First, CourseDREAM outperforms all other competing approaches we tested for course recommendation. Second, we get better results for CS courses than all courses. This was expected as the majority of courses CS students take are within the department; students take non-departmental courses in a less structured manner, making their recommendation harder.

Additionally, we compute the percentage of instances where we make more than one relevant recommendation, which is 58.35%. In 64.5% of the instances, we have more than one correct recommendation of CS courses using the CourseDREAM model. This provides us with some guarantees that the generated recommendations are (even partially) relevant for the majority of the students, and so they can be further used for enrollment prediction. CourseBEACON and Scholars Walk are the next best-performing approaches.

Student Enrollment in Courses

The performance results of the methods for all courses and CS departmental courses appear in Table 2. First, our proposed PfR(ScholarsWalk) has achieved the lowest MAE for all courses and for only over the CS courses in the test set. The majority of the courses are under-predicted by a small number. However, there are a few courses that have significantly high over-prediction errors. We can better understand the per-course performance in Fig. 4. From the direct methods, we observe that LSTM_R performs better for all the courses, but SVM performs better over only CS courses. However, PriorEnr performs better than (overall) or comparable to (for CS courses) these direct methods.

Second, indirect methods (PfR, PfC(LR)) tend to under-predict student enrolment, which can result in fewer course

	all courses			CS courses only		
	MAE	MAE+	MAE-	MAE	MAE+	MAE-
PfR(LSTM)	7.18	24.7 (66)	4.4 (387)	13.82	21.1 (42)	11.3 (99)
PfR(SchWalk)	**5.73**	16.5 (81)	3.6 (367)	**10.68**	16.0 (49)	8.4 (91)
PfR(CBEACON)	6.50	32.2 (47)	3.6 (412)	12.26	23.6 (40)	8.7 (97)
PfR(CDREAM)	6.13	22.2 (64)	3.6 (393)	12.33	18.7 (54)	8.9 (88)
AvgAllPrEnr	13.37	14.7 (413)	3.5 (52)	30.42	34.0 (126)	7.7 (19)
AvgPrEnr	10.64	12.0 (406)	1.8 (49)	26.54	28.3 (135)	5.0 (10)
PriorEnr	6.40	10.0 (269)	3.2 (84)	15.87	18.9 (115)	6.3 (22)
LSTM_R	7.71	9.5 (333)	6.3 (66)	20.14	23.7 (111)	10.8 (28)
RandomForest	11.67	12.3 (424)	5.7 (35)	26.75	29.9 (126)	7.9 (17)
SVM	11.73	11.9 (431)	9.6 (34)	14.99	15.9 (119)	11.1 (26)
GPR	10.24	13.8 (325)	3.1 (90)	22.88	26.3 (123)	5.2 (19)
PfC(LR)	7.96	9.8 (20)	8.1 (435)	18.22	8.8 (13)	20.0 (127)

Table 2: Performance comparison for predicting student enrollment for all courses (left) and CS courses (right). The parenthesis denotes the number of courses over- or under-predicted.

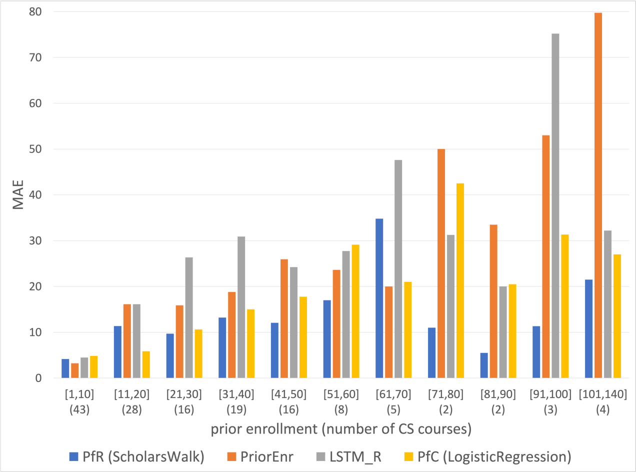


Figure 4: *MAE for best-performing proposed and competing approaches for CS-only courses. The x-axis shows the prior enrollment range with the number of courses in parenthesis.*

sections or offerings than needed. On the other hand, direct methods tend to over-predict enrolment, in particular for CS classes, resulting in excess allocation of resources to accommodate these extra students which might be wasted. PfR(ScholarsWalk) better balances these two scenarios.

Third, we observe higher errors for student enrollment prediction for CS courses than for all courses. The main reason is that most non-departmental courses are taken fewer times and they are predicted fewer times. They also have less room for under-predictions. The average enrollment in all the CS departmental courses of target semesters is around 44 which is much higher than the average enrollment (8) in non-CS courses. So, it is understandable that we have higher under-prediction errors for CS courses.

Limitations Since some non-CS courses are taken by a smaller number of students in prior terms, it is difficult to predict the behavior of students in these courses. There are also a lot of unpopular or specialized CS courses that are taken by a few students (see Fig. 4). The highest errors appear for courses taken by many students, as these courses have more room for under-prediction while they are also popular and might be over-predicted. PfR(ScholarsWalk) though manages to perform the best in these courses. The worst recall corresponds to the worst MAE for PfR(LSTM), while Scholars Walk behaves better than the other PfR methods. Scholars Walk is the only method that takes directly into account course popularity and penalizes it before recommending courses. Next, this type of model might not perform well on courses with recent changes in popularity. Updating the model regularly every semester could help alleviate this issue. Another limitation of our work is that our PfR model can not predict enrolment for new courses. Finally, in future work, we will explore how to include the students filtered out in the preprocessing stage.

Conclusion

In this paper, we propose to use a two-step approach to predict student enrollment in courses. We use course recommendations as a student enrollment predictor for the next semester. We test four different course recommendation models with our framework. Our experimental evaluation with the real-world course enrollment data demonstrates that our proposed course recommendation model PfR(Scholars Walk) performs better than existing competing approaches by providing lower errors. Overall, our proposed approach can be used to provide impactful administration support to a department for resource allocation based on students' needs.

Acknowledgments

This work was supported by the Knight Foundation School of Computing and Information Sciences at FIU.

References

Aksenova, S. S.; Zhang, D.; and Lu, M. 2006. Enrollment prediction through data mining. In *2006 IEEE International Conference on Information Reuse & Integration*, 510–515. IEEE.

Al-Badarenah, A., and Alsakran, J. 2016. An automated recommender system for course selection. *International Journal of Advanced Computer Science and Applications* 7(3):166–175.

Bendakir, N., and Aïmeur, E. 2006. Using association rules for course recommendation. In *Proceedings of the AAAI workshop on educational data mining*, volume 3, 1–10.

Biswas, A.; Ke, Y.; Khuller, S.; and Liu, Q. C. 2023. An algorithmic approach to address course enrollment challenges. *arXiv preprint arXiv:2304.07982*.

Cecez-Kecmanovic, D. 2002. The discipline of information systems: Issues and challenges. *AMCIS 2002 Proceedings* 232.

Chen, M.-Y., and Chen, B.-T. 2015. A hybrid fuzzy time series model based on granular computing for stock price forecasting. *Information Sciences* 294:227–241.

Chen, F., and Cui, Y. 2020. Utilizing student time series behaviour in learning management systems for early prediction of course performance. *Journal of Learning Analytics* 7(2):1–17.

Chen, Y. A.; Li, R.; and Hagedorn, L. S. 2019. Undergraduate international student enrollment forecasting model: An application of time series analysis. *Journal of International Students* 9(1):242–261.

Chniti, G.; Bakir, H.; and Zaher, H. 2017. E-commerce time series forecasting using lstm neural network and support vector regression. In *Proceedings of the international conference on big data and Internet of Thing*, 80–84.

Davidson, J. S. 2005. *Enrollment Prediction at a Texas Baptist University*. Baylor University.

Deb, C.; Zhang, F.; Yang, J.; Lee, S. E.; and Shah, K. W. 2017. A review on time series forecasting techniques for building energy consumption. *Renewable and Sustainable Energy Reviews* 74:902–924.

Egbo, M., and Bartholomew, D. 2018. Forecasting students' enrollment using neural networks and ordinary least squares regression models. *Journal of Advanced Statistics* 3(4).

Fakhrazari, A., and Vakilzadian, H. 2017. A survey on time series data mining. In *2017 IEEE International Conference on Electro Information Technology (EIT)*, 476–481. IEEE.

Haiyang, L.; Wang, Z.; Benachour, P.; and Tubman, P. 2018. A time series classification method for behaviour-based dropout prediction. In *2018 IEEE 18th international conference on advanced learning technologies (ICALT)*, 191–195. IEEE.

Hopkins, D. S. 1981. *Planning models for colleges and universities*. Stanford University Press.

Huarng, K.-H., and Hui-Kuang Yu, T. 2013. Forecasting regime switches to assist decision making. *Management Decision* 51(3):515–523.

Ismail, Z., and Efendi, R. 2011. Enrollment forecasting based on modified weight fuzzy time series. *Journal of Artificial Intelligence* 4(1):110–118.

Jiang, W.; Pardos, Z. A.; and Wei, Q. 2019. Goal-based course recommendation. In *Proceedings of the 9th international conference on learning analytics & knowledge*, 36–45.

Kardan, A. A.; Sadeghi, H.; Ghidary, S. S.; and Sani, M. R. F. 2013. Prediction of student course selection in online higher education institutes using neural network. *Computers & Education* 65:1–11.

Kaur, P.; Polyzou, A.; and Karypis, G. 2019. Causal inference in higher education: Building better curriculums. In *Proceedings of the Sixth (2019) ACM Conference on Learning@ Scale*, 1–4.

Khan, M. A. Z., and Polyzou, A. 2023. Session-based course recommendation frameworks using deep learning. In *Proceedings of the 16th International Conference on Educational Data Mining*, 269–277.

Kraft, J., and Kraft, A. 1977. Determinants of common stock prices: A time series analysis. *The journal of finance* 32(2):417–425.

Lavilles, R. Q., and Arcilla, M. J. B. 2012. Enrollment forecasting for school management system. *International Journal of Modeling and Optimization* 2(5):563.

Lee, M. H.; Efendi, R.; and Ismail, Z. 2009. Modified weighted for enrollment forecasting based on fuzzy time series. *Matematika: Malaysian Journal of Industrial and Applied Mathematics* 67–78.

Lee, D. 2020. *A Classy Affair: Modeling Course Enrollment Prediction*. Ph.D. Dissertation.

Lim, B., and Zohren, S. 2021. Time-series forecasting with deep learning: a survey. *Philosophical Transactions of the Royal Society A* 379(2194):20200209.

Mehtab, S., and Sen, J. 2020. A time series analysis-based stock price prediction using machine learning and deep learning models. *International Journal of Business Forecasting and Marketing Intelligence* 6(4):272–335.

Mondal, P.; Shit, L.; and Goswami, S. 2014. Study of effectiveness of time series modeling (arima) in forecasting stock prices. *International Journal of Computer Science, Engineering and Applications* 4(2):13.

Morsy, S., and Karypis, G. 2019. Will this course increase or decrease your gpa? towards grade-aware course recommendation. *arXiv preprint arXiv:1904.11798*.

Nandeshwar, A., and Chaudhari, S. 2009. Enrollment prediction models using data mining. *Retrieved January* 10:2010.

Nguyen, T.; Vu, N.; and Ly, B. 2022. An approach to constructing a graph data repository for course recommendation based on it career goals in the context of big data. In *2022 IEEE International Conference on Big Data (Big Data)*, 301–308. IEEE.

Pardos, Z. A., and Jiang, W. 2020. Designing for serendipity in a university course recommendation system. In *Pro-*

ceedings of the tenth international conference on learning analytics & knowledge, 350–359.

Pardos, Z. A.; Fan, Z.; and Jiang, W. 2019. 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.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; et al. 2011. Scikit-learn: Machine learning in python. *the Journal of machine Learning research* 12:2825–2830.

Polyzou, A.; Nikolakopoulos, A. N.; and Karypis, G. 2019. Scholars walk: A markov chain framework for course recommendation. *International Educational Data Mining Society*.

Saini, P., and Jain, A. K. 2013. Prediction using classification technique for the students' enrollment process in higher educational institutions. *International Journal of Computer Applications* 84(14).

Shahvaroughi Farahani, M., and Razavi Hajiagha, S. H. 2021. Forecasting stock price using integrated artificial neural network and metaheuristic algorithms compared to time series models. *Soft computing* 25(13):8483–8513.

Shao, L.; Ieong, M.; Levine, R. A.; Stronach, J.; and Fan, J. 2022. Machine learning methods for course enrollment prediction. *Strategic Enrollment Management Quarterly* 10(2):11–29.

Shao, E.; Guo, S.; and Pardos, Z. A. 2021. Degree planning with plan-bert: Multi-semester recommendation using future courses of interest. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 14920–14929.

Slim, A.; Hush, D.; Ojah, T.; and Babbitt, T. 2018. Predicting student enrollment based on student and college characteristics. *International Educational Data Mining Society*.

Soltys, M.; Dang, H.; Reilly, G. R.; and Soltys, K. 2021. Enrollment predictions with machine learning. *Strategic Enrollment Management Quarterly* 9(2):11–18.

Ujkani, B.; Minkovska, D.; and Stoyanova, L. 2021. A machine learning approach for predicting student enrollment in the university. In *2021 XXX International Scientific Conference Electronics (ET)*, 1–4. IEEE.

Wang, H.; Wang, H.; Guo, J.; and Feng, H. 2014. A fuzzy time series forecasting model based on yearly difference of the student enrollment number. In *2nd International Conference on Soft Computing in Information Communication Technology*, 172–175. Atlantis Press.

Ward, J. 2007. Forecasting enrollment to achieve institutional goals. *College and University* 83(3):41.

Watkins, A., and Kaplan, A. 2018. Modeling in r and weka for course enrollment prediction. *International Journal of Institutional Research and Management* 2(1):1–17.

Wong, C. 2018. Sequence based course recommender for personalized curriculum planning. In *International Conference on Artificial Intelligence in Education*, 531–534. Springer.