A Classy Affair: Modeling Course Enrollment Prediction

The Harvard community has made this article openly available. Please share how this access benefits you. Your story matters

A Classy Affair:
Modeling Course Enrollment Prediction

Dianne Lee

Harvard University

Cambridge, MA

Supervisor

Professor Jim Waldo

In partial fulfillment of the requirements for the degree of

Bachelor of Arts in Computer Science
Mind, Brain, and Behavior Track

April 10, 2020

Abstract

The problem of course enrollment prediction has many implications in the determination of university policy. Namely, logistic concerns around course planning cause many universities, Harvard among them, to consider moving away from allowing students a "shopping period" prior to finalizing their courses. This thesis addresses and evaluates these concerns through the development of several predictive models to forecast course enrollment figures in the context of section allocation.

Much of the prior research in enrollment prediction does not provide sufficiently rich evaluation of models fit for the problem of section allocation. Even earlier work in predicting Harvard section allocation does not model the problem comprehensively, not accounting for significant features of the system such as variance in departmental section sizes. Moreover, the existing literature only compares one type of machine learning model and baseline for specific test sets. Previous research does not address the divide between new courses, which inherently have a smaller feature set, and continued courses, other than to note differences in accuracy. Furthermore, existing models in the literature only utilize quantitative features to model course attributes, and do not consider the qualitative aspects considered by students and human baseline predictions.

This thesis addresses these gaps in the literature and performs experiments on updated data in order to develop and evaluate

an updated and comprehensive approach to predicting enrollment in the context of Harvard's section allocation problem. Four evaluation metrics are developed based on previous work as well as qualitative interviews. Multiple machine learning approaches, automatic baselines, and human baselines are implemented and compared. New and continued courses are differentiated within the modeling process in order to analyze and retain their unique attributes. The qualitative feature of course topic relevance is approximated through natural language processing.

We found that for existing courses, both ML models and one automatic baseline outperformed the human baseline. The Random Forest model displayed the best performance across nearly all evaluation metrics, with past enrollment being the most significant feature. Within new courses, no models showed significant improvement over the human baselines across a sufficient number of metrics. However, in terms of predicting raw enrollment, all models outperformed the human and automatic baselines despite their limited feature set.

The findings from this thesis support the inclusion of predictive learning models in Harvard's course enrollment prediction and section allocation process. Although the results do not indicate that a machine learning model should replace human predictions entirely, particularly in the case of newly offered courses, predictive models offer insights and advantages over human baselines. Future work should consider further optimization of these models and the incorporation of a more complete feature set.

Acknowledgements

I would like to thank the following people who helped make this research possible:

First and foremost, my thesis supervisor, Professor Jim Waldo, for his unending support, guidance, and mentorship;

Professor Stuart Shieber, for his counsel as a thesis reader and for sharing with me his experience and expertise;

Professor Bernhard Nickel and Dean Lisa Laskin, for sharing their time and insight into this topic;

Professor Yiling Chen and Dr. Daniel Weinstock, for supporting me in this wonderful field of research as my A.B. and S.M advisors respectively;

The FAS Registrar's Office and the Office of Undergraduate Education, for access to and compilation of data;

The Computer Science department, the Mind, Brain, and Behavior program, and the Institute for Applied Computational Science for providing me with countless opportunities at their intersection and always challenging me intellectually;

My friends and roommates, for their encouragement and patience;

And finally, my parents and brother for their eternal love and support.

Contents

List of Figures

Introduction

1.1   Background

During a Faculty of Arts and Sciences meeting in Spring 2018, Dean of the College Rakesh Khurana presented a proposal to introduce a pre-registration system to the undergraduate course selection process [11]. After months of debate, the faculty voted to table the issue until at least 2022 [3, 4]. However, the issue of course registration continues to be an open one. The same meeting established a standing faculty committee to study the current course registration system and explore alternatives [10, 4]. If the committee supports instituting pre-registration, it could result in a proposal to overturn the current system in Spring 2022.

This is not the first time the issue of undergraduate course registration has caused significant disagreement amongst the faculty. A similar proposal was suggested in Fall 2002 by Dean of the Faculty of Arts and Sciences William Kirby [19]. Many of the reasons in favor of pre-registration remain the same, centering around the logistical difficulties of the current course registration system.

The current process for undergraduate course enrollment includes a "shopping week" at the beginning of the semester where students may attend several different courses before finalizing enrollment. The formal course registration is typically set for the fifth day of classes, after which actual enrollment numbers are generally set. This policy is in contrast to those at many other universities, where students are required to enroll in the courses they will take before the semester begins [23].

Proponents of shopping week support the system because of the benefits and flexibility it provides students and faculty. Students are able make more informed decisions on their courses for the semester based on full information, including finalized syllabi, the first couple of lectures, and introductory assignments if available [23]. Faculty are also able to develop their courses further based on initial student

attendance and feedback. Opponents of shopping week focus on three reasons, according to a report released by the Committee on Course Registration. First, many faculty find it difficult to begin serious instruction or section/lab scheduling until attendance is stabilized the second week of courses — this is particularly a significant loss for courses that only meet once a week. Second, the uncertainty around enrollment numbers often causes undue stress at the directive level. Teaching fellows often don't know for sure which courses they will teach, as courses are generally assigned a TF for approximately every fourteen students. Faculty may have to change course plans in order to accommodate a larger or smaller class than expected. Administrators scramble to assign rooms and spaces during shopping week. Third, students are often unable to finalize their schedule and report stress when negotiating varied procedures for lotteried courses [9].

This thesis treats the assignment of lotteries as a separate topic, as it is less about the establishment of shopping week as a whole, and it seems clear from student interviews that most individuals agree that the benefits of shopping week outweigh any inconveniences. We turn then to the logistic concerns related to shopping week, namely that much planning cannot happen without knowing the exact enrollment numbers for courses. It is then clear that these concerns can be addressed if we are able to predict the approximate enrollment to a certain degree of accuracy.

This thesis proposes to do this through developing an method to predict enrollment in courses based on available data, such as historical enrollment and CUE scores. The committee report also recommends that "sophisticated algorithms" based on available data be "created and deployed as soon as possible [9]." To this end, we outline relevant background and prior research in the following sections. We describe the data and the methodology we used in more detail in Section 2. In Section 3, we outline the development process and theory behind the final models we compared. Section 4 discusses evaluation methods and baselines, as well as findings. We conclude with a discussion of the implications associated with the most significant findings and broader recommendations in Section 5.

1.2   Course Enrollment and TF Allocation

This section outlines the current state of the course enrollment and teaching fellow allocation process at the College. This information was compiled through interviews with the chair of the faculty committee researching course registration, Professor

Bernhard Nickel; the head of the faculty subcommittee researching enrollment algorithms, Professor Jim Waldo; and Dean of the Office of Undergraduate Education Lisa Laskin. See A.1 for a glossary of related terms.

The course enrollment timeline remains consistent from a student perspective. During shopping period, students may attend various different courses and lottery for those with enrollment caps. By the course registration deadline, typically a week into the semester, students are required to register for courses on my.harvard. Students may add or drop courses by the add/drop deadline, the fifth Monday of the term. Withdrawal is also available as an option for students until the seventh Monday of the semester [10].

From the perspective of teaching fellow allocation, this process becomes more complex and decentralized. The timeline for teaching fellow allocation is generally standard across the board. Departments make preliminary student enrollment predictions for each course, determining an initial estimate for the number of teaching fellows required per course. These estimates are submitted to the Office of Undergraduate Education. The OUE then submits their own preliminary enrollment and TF predictions based on these departmental estimates. Based on section sizes for each course and differences in predicted and actual enrollment, teaching fellow allocations may have to be readjusted after the registration deadline [personal communication, 2019][1].

The process becomes more complicated at the implementation level because of departmental differences in section size assignment, enrollment prediction, and allocation needs. Each of these categories may vary significantly by department. In the School of Engineering and Applied Sciences (SEAS), the universal section size for courses is set at fifteen students. In other words, courses are assigned approximately one TF per fifteen students. Course enrollment is predicted to be the enrollment the previous time a particular course was offered. TFs are assigned a teaching load of one course each [2].

Comparatively, in the Philosophy department, target section sizes are generated based on the type of course. Logic courses heavy on problem sets are assigned a section size of 10. Writing intensive courses with more papers to grade are assigned a section size of 15. More general courses have a target section size of 18. The Director of Undergraduate Studies inputs estimates of enrollment by course based on

---

[1]Interview with Dean Lisa Laskin on March 4, 2019.
[2]Interviews with Professor Jim Waldo.

factors such as previous year's enrollment and CUE guide scores, instructor appeal, relevance of subject, longevity of offering, and any anomalies such as popular courses outside the department that might overlap with the schedule. TFs are also required to teach two sections each term. This works out similarly to the SEAS approach when there are an even number of sections assigned to a course, so each TF can be assigned two sections in the same course. However, when there are an odd number of sections assigned to a course, TFs may be required to "split fifths" and teach two courses during the term, which increases their workload considerably [3]. While the philosophy case generally applies to courses in the Arts and Humanities department, section size assignment, enrollment predictions, and TF teaching load considerations vary significantly across departments and across individual subjects.

There are also further considerations, such as the fact that some departments guarantee graduate students TF positions for the first few years of their programs, during which time they would be assigned positions independent of course enrollment numbers. However, we consider these individual differences to be outside the scope of this paper as they serve to simplify the issue – guaranteed TF assignments may be determined after enrollment-based assignments.

Based on the available information, we determined the following issues to be most significant in addressing the problem of accurate course enrollment prediction for the purpose of more efficient TF allocation.

1. **Prediction Accuracy**: A model that is foremost as accurate as possible to predicting the actual enrollment in each course is an initial requirement. Such a model would standardize and quantify factors currently used for prediction, such as past year's enrollment and CUE scores, as well as engineer other features that might capture enrollment trends.

2. **Prediction Relevancy**: It is important to consider not only the objective prediction accuracy but also the relevance to the net goal of optimizing TF allocation. A relevant model would consider relative evaluation metrics, such as the individual section sizes per course, as well as minimizing inaccuracies in predicting "splitting fifths" in relevant departments.

3. **Algorithm Applicability**: Another consideration after a model is specified is its ability to be implemented and utilized in the course enrollment system.

---

[3]Interview with Professor Bernhard Nickel on February 26, 2019.

Towards this goal, a model should use data available universally across courses and departments, and be designed in such a way that it can output a prediction for any course at the College, past, present, or future.

This paper attempts to address each of these issues, and starts by locating previous literature in this area.

1.3   Related Work

Course enrollment prediction through machine learning is a relatively understudied field. While there are a number of papers that have looked into this type of problem, most of the literature focuses on predicting overall enrollment for the student body as a whole. Britney compares two such models: (1) Markov chain models and (2) circuitless flow networks. Markov chain models use cross-sectional data about the student population and historical yields to predict total student body population. Flow networks rely on longitudinal grade progression data and past student performance to model more specific predictions. [6]. Hopkins and Massy extend Britney's research and review another simpler category of enrollment model: the grade progression ratio (GPR) method [14]. GPR determines the ratio of students that progress between grades in order to inform their predictions. Hopkins and Massy claim that these categories of models can also be applied at the department or course level but do not actually implement such a model. The general nature of these models prevent them from having much prediction relevancy to the TF allocation problem. The GPR method in particular relies on entire classes of students that progress through school together; in course enrollment, where students individually select their own classes, this sort of model has little basis. In addition, the circuitless flow network introduced by Britney relies on sensitive student data, such as performance, which we have no access to and should not expect to incorporate in a final model implemented for the university. In other words, the model has no algorithm applicability for our case. However, we revise and extend the Markov model in order to capture and compare the overall enrollment prediction approach.

Some research focuses on predicting course enrollment more specifically. Balachandran and Gerwin developed three variable-work models used to predict course enrollments: the work model, the eligible-work model accounting for prerequisites, and the eligible work model with program requirements [1]. Each model starts with individual student course predictions and incorporates additional data, such

as course prerequisites and degree program requirements. Through testing on graduate courses, the authors found high error rates and were not able to confirm the validity of these models. While we incorporate the general idea of this approach in our Markov model, we are limited by the data we have and do not implement these particular models to ensure algorithm applicability further down the line.

Kraft and Jarvis took a different approach in developing their adaptive model for course enrollment (2005). They identified significant characteristics based on a variety of student data, including major and transfer status, in order to fit conditional probability models to clusters of similar students rather than the entire student body. While this paper showed some promising results, its reliance on an abundance of individual student attributes, some of them potentially sensitive, did not translate well to our limited per-student dataset. We looked instead to Johnson and Strohkorb's approach to using logistic regression to predict course enrollment on a per student basis [15]. The authors extracted students' course histories and fit models for specified courses and semesters. The authors found large discrepancies in accuracy, especially for newly offered courses. In order to address this issue, we limited our logistic regression model to courses offered more than once

While each of these papers tackle some aspect of course enrollment, they focus almost solely on enrollment as an evaluation metric. Only Kraft and Jarvis address a different problem, that of seat allocation, which still uses enrollment as the evaluation metric. This exclusive evaluative focus on enrollment prevents these models from having prediction relevancy to the problem of TF allocation. The study that was most relevant to our research was the final report from the Fall 2003 iteration of Computer Science 96 (System Design Projects). Students in the course investigated the same question of keeping shopping week versus instituting early registration at Harvard College and generated a report with their results and recommendations [2]. The report applied a Support Vector Machine learning layer to features extracted from historical course enrollment data from Fall 1993 to Spring 2003. The authors found that this resulted in a significantly lower error than a human or automatic baseline. However, the authors did not have access to departmental predictions at the time and approximated a human baseline through a series of surveys. Additionally, the report used section allocation information standard to SEAS instead of on a departmental basis. We looked to verify and improve these results on a more comprehensive dataset and implement an updated machine learning system based on new information.

We adopted many of the procedures outlined in the CS96 report with several key differences. (1) We trained and tested our models on a completely new dataset, from Spring 2004 to Fall 2019, given data availability. (2) We revised our feature set based on data availability and incorporated measures for qualitative attributes used in making departmental predictions. (3) We updated our evaluation metrics and data based on new knowledge about how sectioning is conducted in departments outside of SEAS. (4) We separated out new and existing courses in order to more accurately compare predictive power between the two groups. (5) Our work leverages increased computational power to compare several different models, in addition to the final Support Vector Machine model outlined in the report trained on our revised feature set.

Finally, given the relative paucity of past research on course enrollment specifically, we also adopted a couple of models commonly used in machine learning and applied to fields outside of enrollment prediction. We selected a Random Forest Regressor for its flexibility and generalizability [17]. Random forests have been shown to perform well in a diverse set of fields, from genomic data analysis to rainfall estimation [7, 20]. In the field of mineral prospectivity, random forests have been shown to perform better than support vector machines [8].

Given the reduced feature set inherent to newly offered courses, we also adopted a clustering approach. We selected the agglomerative hierarchical clustering method because of its explainability [18]. Hierarchical clustering has been used in prediction problems as an initial step for comparison. In addition to Kraft's per student enrollment model, clustering has been used in several different fields, including as a way to predict consumer preferences in tomatoes. [21]

Data

A machine learning model is only as good as the data it is fed. Accordingly, a large portion of this thesis is dedicated to the course enrollment data used to train our later models.

Our data was obtained from three main sources. Historical course enrollment information, course evaluation data, and per-student enrollment data were procured from the Registrar of the Faculty of Arts and Sciences. The historical course enrollment and per-student enrollment information was split into two pieces: Spring 2004 - Fall 2014 and Spring 2014 - Fall 2019. This is due to the transition from manual student study cards to the online my.harvard course shopping carts in 2014 [5]. Course evaluation data was taken directly from student reviews in the CUE Guide [13]. Available course enrollment departmental and official predictions were sourced from the Office of Undergraduate Education. Additional information, including features used to inform departmental predictions and otherwise unavailable enrollment records, were acquired through interviews with members of the Faculty Committee on Course Registration.

Each of these datasets contain information about courses each academic term they are offered. In order to establish a consistent terminology, we define a "course" as the general course offering across all available semesters it was offered, e.g. "COMPSCI-105." We define a "course instance" as the specific course/term pairing, e.g. "COMPSCI-105, Fall 2019."

The above data can be split into four main categories: Historical Course Enrollment Information, Course Evaluation Data, Student Enrollment Data, Course Enrollment Predictions The categories of data and information available through each are described in more detail below:

Historical Course Enrollment Information The course enrollment data is split into pre-online (Spring 2004 - Fall 2014) and post-online (Spring 2014 - Fall 2019). The post-online data was combined with per-student records, so only the data unique to each course is discussed here. Each dataset contained entries about each course offered at FAS in the included terms. There were several potential identifiers associated with each entry such as the course catalog number, course title, subject, and course number. The most cohesive identifier across datasets was found to be a concatenation of the subject (an abbreviation for the subject name, e.g. 'AFRAMER' for African-American Studies) and the course number. Each entry specified additional information such as department, an anonymized instructor ID (a unique set of 8 digits mapped to the original instructor HUIDs to maintain privacy), and binary classifiers for General Education or Core courses. The original pre-online dataset contained 43,758 course instance entries while the post-online dataset contained 447,891 course instance/huid mappings. Both datasets were initially reformatted as course entries, storing course instance data as terms. The pre-online dataset consisted of 11,471 courses and the post-online dataset consisted of 10,437 courses.

Student Enrollment Data Similarly to the course enrollment data, the student enrollment data is split into pre-online (Spring 2004 - Fall 2014) and post-online (Spring 2014 - Fall 2019). The pre-online data consisted of course identifying information (of which we used the concatenation of the subject and course number as described above) as well as anonymized student IDs (a unique set of 8 digits mapped to the original student HUIDs to maintain privacy) enrolled in the course. Together, this consisted of 790,322 entries. The post-online student records were combined with the course enrollment data to make one entry per student in each course by academic semester, a total of 447,891 entries. These records were cross-referenced and aggregated with the course enrollment data as described in the Data Cleaning section.

Course Evaluation Data The course evaluation data consists of averaged student evaluations for available courses by academic term and was obtained from the CUE Guide and compiled by the Registrar's office. The evaluations recorded were the course mean (the averaged value across all student evaluations of the overall course), the recommend mean (the average student response to the ques-

tion "Would you recommend this course to another student?"), and the workload mean (the average student evaluation of the hours spent per week on the course) [13]. Not all courses were represented in the dataset for a number of reasons: Smaller courses often do not have any student evaluations, some courses do not publish their results, and other courses may not be represented due to data reduction steps discussed in the following section. The available information was aggregated with the historical course enrollment data as described in the Data Cleaning section. There were a total of 31,210 entries in the course evaluation dataset covering the terms Spring 2006 - Fall 2018.

Course Enrollment Predictions The course enrollment prediction data consisted of departmental and official predicted enrollments obtained from the Office of Undergraduate Education for the academic year 2018-2019. The prediction data covered four stages of the enrollment process for each available course: (1) preliminary departmental enrollment predictions, (2) preliminary OUE enrollment predictions after the departmental predictions are submitted, (3) OUE enrollment allocations after the study card deadline, (4) final enrollment numbers. For each of these stages, the information extracted consisted of the predicted course size, predicted section size, and predicted number of sections. In the final enrollment stage, these records translated to the final enrollment, average section size, and final number of sections. Some courses were not represented in the OUE predictions, most notably courses in the Engineering and Applied Sciences department, which runs its own course registration system. After interviews with members of the Faculty Committee on Course Registration as well as Dean Laskin from the OUE, we found that SEAS predicts a section size of 15 for its courses, and predicts the current year's enrollment in a course solely based on the past year's enrollment. This data was imputed in the data cleaning process, described in further detail in the next section. In total, the OUE dataset consisted of 1573 records.

In order to more easily incorporate the separate datasets, we aggregated the individual information by course identifier/student identifier. These were stored as CourseInfo and StudentInfo objects containing the data for each course and student as well as functions to easily access and modify their information (See A.2 for more details). Given the separate ways the data was recorded as well as the uncatalogued

revisions made to course information during the transition to my.harvard, the data aggregation could not be perfect. The decisions and trade-offs made in cleaning the data are described in more detail in the following section.

2.1 Data Cleaning

In order to aggregate enough relevant features of the existing datasets while minimizing any loss of information, a large portion of this thesis was spent on data cleaning. There were two main issues with the data. First, the fields were not standardized across each dataset. For example, the Classics department was encoded as 'CLASSICS' in some datasets and 'CLASSICSSTDY' in others. Second, some data points were irrelevant to our problem and would be misleading if included in our models. For instance, we should disregard cross-registered courses as our problem is scoped to courses and students at the College.

We addressed both of these problems in our cleaning process, and outline our approach as well as the trade-offs necessary in the following sections.

2.1.1 Field Standardization

One of our main concerns was making sure fields were standardized across each dataset in order to later generate accurate categorizations and features. The main fields processed were the following:

Course Identifier   The most consistent course identifier was found to be a concatenation of the subject and the course number. We standardized the subject name and course number, as well as text in all following fields, to be case-insensitive and solely alphanumeric. Some specific subjects were encoded differently across datasets and were identified and standardized by hand. See A.3 for more detailed information on subject mappings. The final course identifier was the standardized subject and course number concatenated with a dash, e.g. "COMPSCI-50."

Academic Term   Another important identifying field was the academic term. The year and semester (Fall/Spring) were encoded differently across all datasets, but were standardized to the form YYYYT, where T was 1 for the Spring semester

and 2 for the Fall semester. This was set in order to chronologically order terms (Spring < Fall).

Department   We found that department was significant in encoding features for our model later on, but it was also one of the most unstandardized fields across datasets. Aberrations were determined by hand and either standardized or grouped into reasonable categories. For example, engineering departments were all originally recorded as "Engineering & Applied Sciences" for the most part, except for some subjects in Spring 2014, Fall 2014, and Spring 2015, when the data was most likely being transferred to my.harvard. Interestingly, there were no differences in department encoding in 2007, when SEAS transitioned from being an FAS division to a separate school [16]. To aggregate the encoding inconsistencies, departments with "Engineering" in the title as well as some hand-coded exceptions were mapped to "Engineering & Applied Sciences." More detailed mappings can be found in A.3.

Specific Adjustments   There were a few other specific data adjustments that had to be made for accuracy. In Fall 2012, Spring 2013, and Fall 2013, ECON-1010 was split into ECON-1010a1 and ECON-1010a2. These enrollments were combined into the more general ECON-1010a. In Fall 2013 and Spring 2013, ECON-1123 was recorded as ECON-1123a1. This was combined into the general ECON-1123. In 2013 and 2014 Fall, COMPSCI-50 enrollment was recorded separately for students who took it Sat/Unsat and for a letter grade. These were added and combined to the more general COMPSCI-50.

For more detailed information on the specific mappings used for the standardization process, see A.3.

2.1.2   Data Scoping

After standardizing fields, relevant data was narrowed down from the aggregated information. Cross-registered courses, for example, were excluded because graduate schools assign teaching fellows according to separate systems. Each of the following cases was excluded from the final dataset:

Cross-Registered Courses   In order to scope our problem to the College, cross-registered courses were excluded, as graduate schools generate teaching fellow

assignments according to separate systems. Cross-registered courses were identified by courses starting with an "X." This removed 2,326 courses overall.

General Education Courses Due to the new General Education system, as well as disrupting factors during the transition to the new system, General Education courses were removed. Under the new system, the course lottery is run by the General Education department and thus any predictions generated by our model are irrelevant. General Education courses were defined as those in the eight specified General Education subjects as well as in the new "GENED" subject. Regular departmental courses that have General Education status were retained because they do not fall under the purview of the General Education department. This removed 951 courses.

Individual Seminar Courses Individual seminars are defined to be seminars where the teaching fellow assignment is determined by the department rather than under the College system, specifically Expository Writing and Freshman Seminar courses. Expository Writing courses were identified by courses under the subject "EXPOS" and Freshman Seminar courses were identified by the subject "FRSEM." This deleted 2,396 courses.

Research Courses Courses for independent student or thesis research do not have caps and were excluded. These courses did not have a standard identifier, but we determined a reasonable mapping through information available on course department websites. See A.3 for more detailed information on how research courses were defined. This process removed 4,574 courses.

After the cleaning process, we were left with 8,384 courses in the pre-online course enrollment data, 6,264 courses in the post-online course enrollment data, 10,452 courses in the course evaluation data, and 1,573 courses in the course enrollment predictions.

2.1.3 Data Validation

The standardization and cleaning process outlined above by necessity makes a lot of assumptions and is unable to perfectly aggregate our datasets. We honed the process by comparing validation numbers. Our validation consisted of two steps.

First, we validated the course data available during the academic terms that overlapped between the pre-online and post-online data, namely Spring 2014 and Fall 2014. Ideally, these courses would overlap perfectly, but given the non-standard data storage as well as the information gained and lost in the transition to my.harvard, this was nearly impossible. Originally there were 3,407 courses in the pre-online dataset and 4,047 courses in the post-online dataset. There were 2,766 overlapping courses; 641 courses in the pre-online data didn't exist post-online and 1,281 courses vice versa.

Following cleaning, there were 2,113 courses in the pre-online dataset and 2,154 courses in the post-online dataset. There were 1,611 overlapping courses; 502 courses in the pre-online data didn't exist post-online and 543 courses vice versa. This was around a 75% overlap rate for each dataset, which we found reasonable given the obstacles listed previously.

Second and finally, we ran our cleaning process on all of our available data and compared the results from the pre-online and post-online courses. This resulted in 11,471 courses in the pre-online dataset and 10,437 courses in the post-online dataset. There were 5,214 overlapping courses; 6,257 courses in the pre-online data didn't exist post-online and 5,223 courses vice versa. This also seemed reasonable as there were bound to be older courses that phased out before 2014 as well as newer courses that did not start until after 2014.

Given our validation results, we aggregated our pre-online and post-online course data and cleaned the rest of our datasets. Data was aggregated by course, with course instances being stored as terms each course was offered. During this aggregation, if there were two sets of data available in Spring 2014 or Fall 2014 for a course we simply took the post-online version, as we reasoned the electronic records would be more accurate. We combined all available information for each course and each student. Our final course dataset consisted of 11,153 records and our final student dataset consisted of 69,931 records. We also separated out a test dataset limited to courses offered in academic year 2018-19, which we set as our test year, which contained 2,489 records.

2.2   Exploratory Data Analysis

After we compiled our final dataset, we ran an exploratory data analysis to identify features that might be of interest. Taking a look at the number of years each course

was offered, we can see that the minimum number of times a course was offered was 1, and the maximum was 32 (Figure 2.1). 25 courses were offered 32 times, most of which seemed to be a perennial requirement (e.g. MATH-21A and B, Organic Chemistry). A plurality (4,632) of courses were offered once.
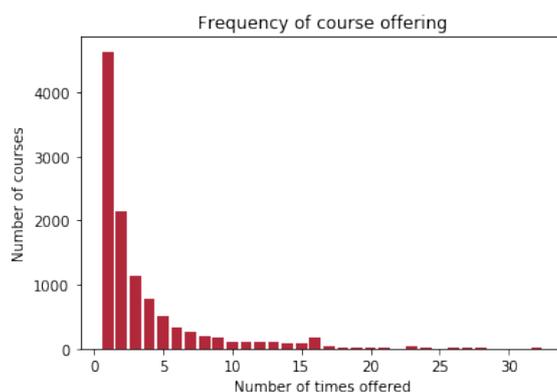
Figure 2.1: Distribution of courses by the number of times each course was offered.

2.2.1 Courses Offered Once

4,632 courses were offered once. Graphing enrollment size, we see that a plurality (2,935) of these courses had less than 10 students enrolled (Figure 2.2). The average enrollment was 11.14.
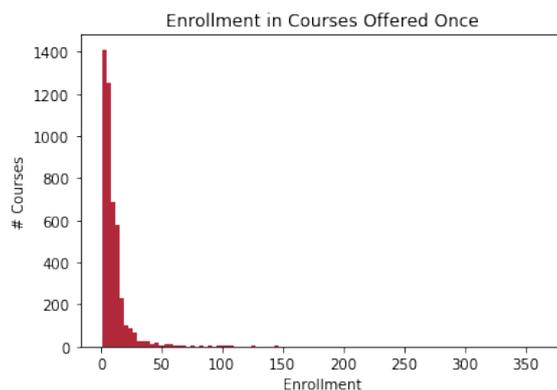
Figure 2.2: Course enrollment distribution across courses offered once.

The largest course was ECON-1152 (Using Big Data to Solve Economic and Social Problems) which had 364 students enrolled in Spring 2019. This course was so popular in its first year because it was publicized heavily and students found the

topic timely and interesting. This indicates the need to capture the relevance of a topic as a feature in our model.

Graphing the number of courses by department, we see that there were 68 departments represented(Figure 2.3). History offered the most courses (483) followed by Government (307) and Romance Languages & Literature (292). We ascribed this distribution to an offering pattern of smaller seminars that phase in and out.

Figure 2.3: Number of courses offered by department, calculated for courses offered once. Department names excluded for readability.

Graphing the average enrollment by department, we see that enrollments stayed around 10 students on average, with a few large outliers. This implies that department may be a significant feature, and clustering by department may yield good results.

Figure 2.4: Average enrollment by department, calculated for courses offered once. Department names excluded for readability.

2.2.2 Continued Courses

In this section we excluded courses that were offered only once. Averaging the enrollment of each course (by each enrolled semester) and plotting against the number of times offered, we see no clear trend as the number of semesters offered increases (Figure 2.5). There are a few outliers, but generally the average enrollment stays consistent around 30. This indicates that an optimal predictive model might focus on identifying those few outliers that are a lot larger than the average class by characteristic features.
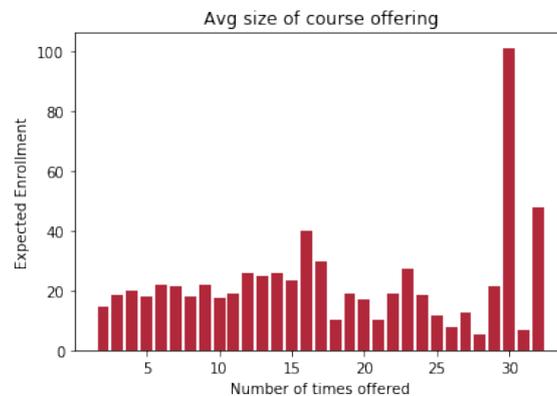
Figure 2.5: Average enrollment by number of times each course was offered. Calculated for courses offered more than once.

Graphing the number of courses by department, we see that there were 76 departments represented (Figure 2.6). History again offered the most courses, 407. Looking into why this might be the case, we see that the number of times a History course was offered is inversely related to the number of such courses. As we saw in the previous section, 483 History courses were offered once. 191 History courses were twice, 88 were offered three times, and the number only decreases as the offering count increases. This supported our previous conclusion that the large number of History courses is due to the constant influx of new seminars that phase out quickly.

Graphing the average enrollment by department, where the ordering of departments across the x-axis is the same, we see that the number of courses and expected enrollment varies wildly across departments (Figure 2.7). We can also see that there is no clear trend in the number of courses offered to the average enrollment per course. This data size discrepancy indicates that we most likely need more features than simply the individual department to cluster and predict course enrollment.
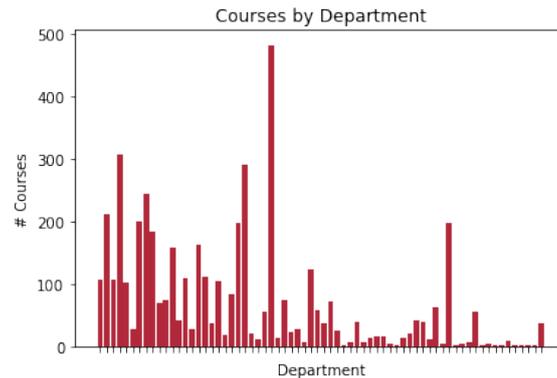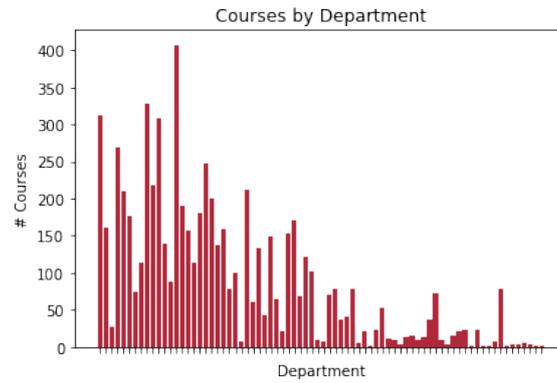
Figure 2.6: Number of courses offered by department, calculated for courses offered more than once. Department names excluded for readability.
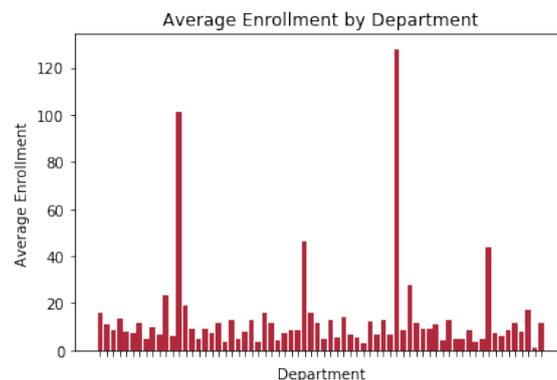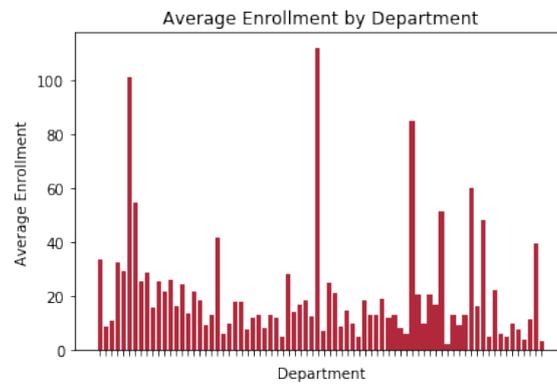

Figure 2.7: Average enrollment by department, calculated for courses offered more than once. Department names excluded for readability.

2.2.3 Student Data
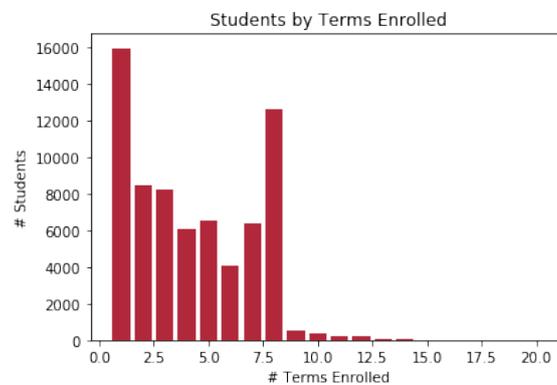
Figure 2.8: Distribution of students across terms enrolled.

25

There were 69,931 student entries. Most students were enrolled in 1 term (15,982) or 8 terms (12,642), as shown in Figure 2.8. This makes sense because most students graduate in 8 terms or 4 full academic years, and temporary students who enroll at the College for one course would only be recorded for 1 term.

The pluarality of students who took only one course is interesting because it indicates that a model based purely on per-student data might not perform as well in predicting future enrollment. Students who enrolled in only one course yield little predictive information about other courses.

Models

3.1 Approach

Our system takes as input a set of raw course data. It then applies feature engineering, extracting aspects of the raw data that we determine to be most relevant. These features are fed into a machine learning layer, which outputs a final predicted enrollment. A diagram of this process can be seen in Figure 3.1. We compare different models in the machine learning layer to determine an optimal predictor.
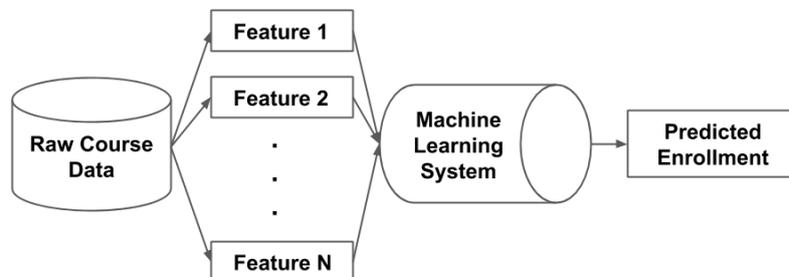
Figure 3.1: Design of our machine learning pipeline.

3.2 Feature Extraction

Features are measurable characteristics derived from individual traits of each observation that aim to capture trends in the data relevant to the model. The selection of independent, informative, discriminating features is critical to model performance.

In order to capture features relevant to enrollment prediction, we consolidated data from prior work as well as qualitative information from interviews with department heads responsible for entering enrollment predictions by hand. In particular, we replicated as many features from the CS96 report as possible to ensure a close final model for comparison. We deviated only when data was unavailable, we were able to engineer better representations or adaptations with updated data, or when we engineered new features based on qualitative interviews and new data. Each feature is designed to produce some value for a predicted (course, term) pair. The feature is derived from data that would be made available before the start of the predicted term. For most fields, this means only using data from the academic term prior to prediction. For example, in order to predict enrollment for courses offered in Fall 2018, we would only use data about courses from Spring 2018 and prior. This ensures that our model results are replicable and our models are able to predict enrollment in any courses offered in the future. The only exceptions to this rule is instructor information and any descriptive course information, such as General Education status. Any features using this data reflects the fact that this information is available to students during the semester the course is offered.

We divide our final features into three buckets: (1) Historical/Raw Features, (2) Course Evaluation Features, (3) Derived Features. Although prior work in course enrollment models have also incorporated features related to course scheduling, instructor ratings, and student descriptors, we did not have access to the data necessary for this analysis and excluded these from our feature list [2].

3.2.1 Features

Historical / Raw Features

1. **Historical Enrollment Feature:** Our historical enrollment feature attempts to report enrollment information about a course from previous terms it was offered. The feature records the enrollment number from the last time a course was offered. If the course if offered for the first time, this number is recorded as 0 to capture the lack of information available. The feature also records the enrollment numbers for a course from the prior three academic years. If this information is unavailable, the feature is imputed with the average enrollment across courses offered that year. For each of these figures, a secondary "flag" value is also reported. Whenever any information is unavailable, the flag value

is set to 1; otherwise the flag value is set to 0.

2. **Categorical Feature:** Our categorical feature attempts to capture different categories a course might fall into. We were more conservative with defining categories, valuing accuracy over number of categories. For example, there was no clear way to identify 100-level courses, as each department sets levels differently. We narrowed down four categories of interest: introductory courses, topics/reading courses, General Education courses, and SEAS courses. Our categorical feature reported four Boolean values for each of these buckets indicating whether a course was identified as each.

3. **Departmental Feature:** Our departmental feature captures average enrollment information for the department in which a course belongs. This is due to the reasoning that the particular enrollment in a course depends in part on the department. The feature records the average enrollment for the last three academic years in a given department, with a secondary flag value indicating if the information is unavailable. For new courses, a feature indicating the average new enrollment for a department was also included. The new enrollment was calculated for courses offered in the last academic year, for terms after 2004 in order to exclude the boundary of our dataset.

Course Evaluation Features

1. **Raw CUE Guide Ratings Feature:** Our raw CUE Guide ratings feature captures available student evaluation information for a course the previous time it was offered. This consists of the average overall rating, recommend rating, and workload. There were two systems available in the CUE Guide data to measure workload. Before Fall 2014, workload scores were collected as a number 1 to 5, each corresponding to a range of hours. Starting Fall 2014, workload hours ware collected as a whole number between 0 to 168. In order to use both sets of information without introducing inaccuracy, we translated post-Fall 2014 workload means to a range between 1 and 5. For each of these ratings, we imputed the data with the mean of all other available ratings, and reported a secondary flag value to indicate data availability. The ratings were then standardized to have a mean of 0 and a standard deviation of 1 because we hypothesized that course enrollment depends on the relative nature of CUE Guide ratings rather than the raw 1-5 number.

2. **Departmental CUE Guide Ratings Feature:** Our departmental CUE Guide ratings feature attempts to capture evaluations across course departments. The feature reports mean CUE ratings (overall, recommend, workload) across departments from the past three academic years. If data is unavailable, a secondary flag value is reported and the feature is imputed with the mean of all other available values for that academic year. The feature is then standardized.

Derived Features

1. **Offering Trends Feature:** Our offering trends feature attempts to capture how often a course has been offered in the last three years, as well as recency of offering. For example, if a course was not offered recently, a student might be more likely to take the course the current term. Additionally, if a course was not offered the previous year, this would have a larger impact on a student's decision to take it than if a course had not been offered three years prior. In order to capture this information, we record the sum of each time a course was not offered in the previous 6 academic terms according to its recency. For example, consider a course with the following offering pattern for the past six terms: offered, not-offered, offered, not-offered, not-offered, not-offered. Then we can first represent the course offering pattern by mapping a 1 to every time the course was not offered and 0 otherwise: 010111. Then the feature is generated by summing the value at each position multiplied by its 1-indexed position to account for recency: $010111 = 0*1+1*2+0*3+1*4+1*5+1*6 = 17$. The feature was standardized across courses in order to reflect the relative rather than absolute value.

2. **Sequencing Feature:** Enrollment in some courses are clearly related to enrollment in others. For example, many students take CS-124 directly after CS-121, as both are requirements for the Computer Science concentration. Our sequencing feature attempts to capture this pattern. First, based on student data available prior to the prediction term for each course, for each course we calculate the probability a student will next take every other course the following term. For each course c for which enrollment is being predicted, the course with the highest probability linked in the previous academic term is labeled the predictor course. The percent change Δp in enrollment of the

predictor course in the prior academic year is calculated. The final feature is reported as Δp*the past enrollment feature of *c*. If any information is unavailable, Δp defaults to 1.

3. **Course Attribute Change Feature:** Our course attribute change feature attempts to capture any change in a course's descriptive information in the past year. Any drastic change may change students' opinions of the course. We focused on change in instructor and General Education status. This was reflected as a Boolean value indicating if the attribute changed between the last time a course was offered to the current offering.

4. **Topic Relevance Feature:** One of the more qualitative features used for departmental predictions of course enrollment is the course topic relevance. While the modeling of this exact process would constitute another paper topic in itself, we engineered a natural language processing feature to capture some aspects of topic relevance. Our feature calculates the simple sum of the term frequency-inverse document frequency (TF-IDF) scores for each of the words in the course title. Term Frequency is a measure of how frequently a word appears in a document and is calculated as $tf_{i,j} = \frac{n_{i,j}}{\sum_k n_{i,j}}$, the number of times a word appears in a document over the total number of words in the document. Inverse Document Frequency measures the importance of a word. Some words, such as "the" may appear many times but are of little importance. IDF weights rare words more heavily while scaling down frequent ones and is calculated $idf(w) = log(\frac{N}{df_i})$, or the log of the number of documents over the number of documents containing the word w. In this case, the course title is treated as a document, while the set of documents is regarded as all other course titles. The TF-IDF score is then generated $w_{i,j} = tf_{i,j} * log(\frac{N}{df_i})$ and the feature computes and returns the sum of scores across each word in the title. The feature was standardized across courses in order to reflect the relative rather than absolute value.

3.2.2 Feature Analysis

In order to analyze our features, we generated a correlation matrix to compare independence (See Figure 3.2). Generally, our features seem to be fairly uncorrelated. However, the features calculated for the past three academic years have high correlation, which implies that in most cases there is probably little variation in enrollment

in recent years. Past enrollment is also highly correlated with the sequencing feature, which makes sense as it is calculated based on past enrollment. The course mean and recommend mean for course evaluation data were also highly correlated, which is to be expected as students who rated courses highly are likely to also recommend them.
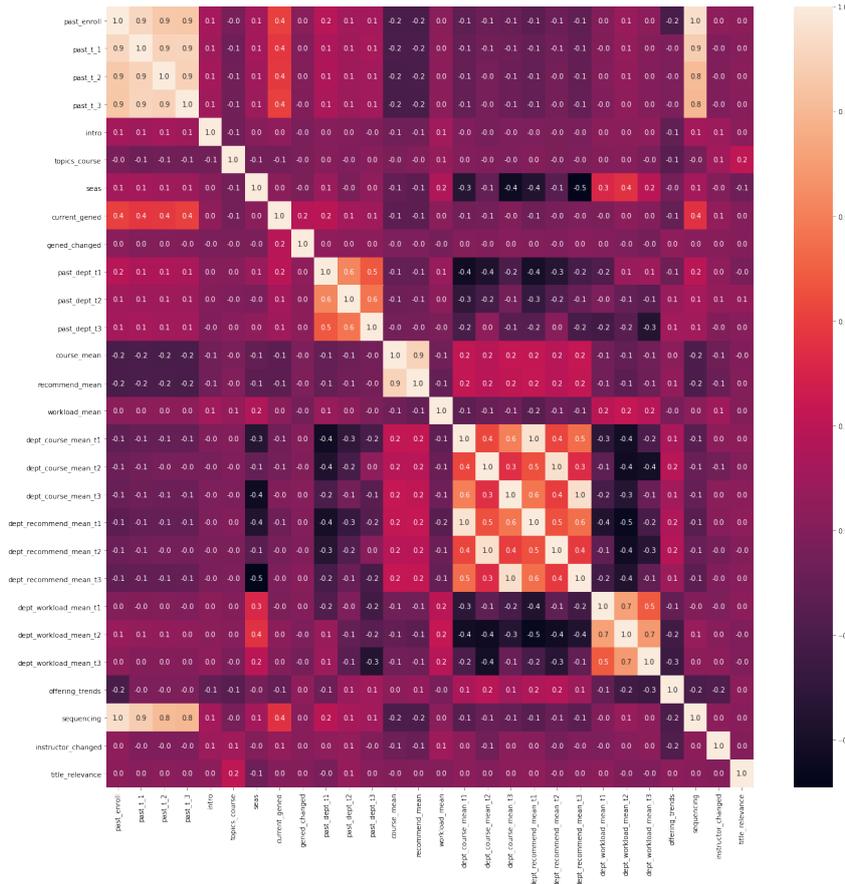
Figure 3.2: A visualization of the feature correlation matrix. Flag values were excluded for readability and relevancy. Data are labeled with their Pearson correlation coefficient.

We also graphed the distribution of each feature (See Figure 3.3). Binary categorical features seemed to be weighted heavily towards zero values, giving us sparse vectors: Most features were outside of the category rather than inside. Enrollment features were also skewed left, indicating that most courses had low enrollment figures. Course evaluation data was more evenly distributed, indicating most students rated courses close to the average.
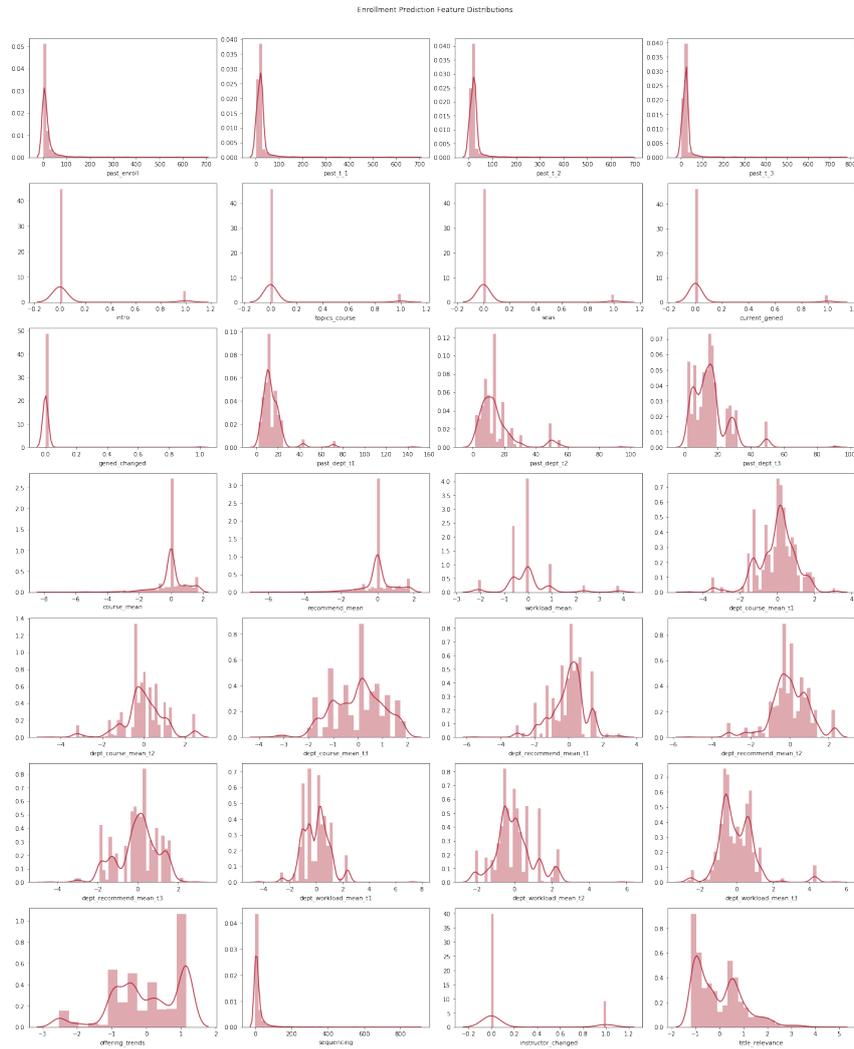
Figure 3.3: Distribution of Features. Feature names are labeled on the x-axis.

3.3   Modeling Approach

Our machine learning system consisted of the training and prediction stages. In the training phase, the system takes in historical data prior to the academic term being predicted. Features are generated from this data according to the process outlined in the previous section. The learning layer takes these features as input. System parameters are tweaked until the system is able to optimally predict future enrollments based on feature values, resulting in some sort of function from feature to predicted enrollment. A successful system should be able to incorporate feature values to generalize to future course enrollment predictions. For example, if all courses that have switched to General Education status have experienced increased

enrollments historically, the machine learning system should learn to use the General Education status change feature to predict a similar effect for future courses.

The prediction phase follows the completion of model training. In this stage, our system is fed a series of test data from the prediction academic year, which is run through the same feature generation process. The trained model is applied to these features to generate predicted enrollments. We can use this output to evaluate the performance of the model, as we do in Section 4.

One issue we had to tackle was how to deal with courses that were offered for the first time and thus had no associated historical features, e.g. past enrollment. Given that a large subset of our features depend on historical data, we decided to split the modeling step into two parts: a model for courses offered more than once (Continued Courses) and a model for courses offered for the first time (New Courses).

3.4 Models for Continued Courses

3.4.1 Single Feature Models

As a preliminary baseline, we implemented two models based on single features. Our **Past Enrollment Model** adopts the automatic baseline from the CS96 report and outputs the enrollment figure the previous time the course was offered [2]. This model is based on the reasoning that courses generally do not deviate from the previous year's enrollment, the idea that underlies SEAS enrollment predictions. Our **Sequencing Model** outputs the sequencing feature for comparison instead.

3.4.2 Per Course Models

Our per course models use course-based information and features to output enrollment predictions.

Random Forest

Our training data consisted of a large number of features, many of which were categorical. This convinced us that a random forest regressor model could be appropriate, as random forests are less sensitive to sparsity than generalized linear models [17].

A random forest is an ensemble model that trains via decision trees. A decision tree is a simple learning algorithm that considers every possible "split" in the data grouped by their features. The decision tree chooses the split that minimizes the overall mean squared error. Each side of this split now becomes a "branch." The decision tree then repeats this process on each branch until it reaches some end criterion, usually a maximum depth for the tree or a minimum number of observations included in each split. The decision tree makes predictions by classifying the new observation into a split and outputting the mean. A random forest further addresses the overfitting problem native to decision trees by building a large number of different decision trees based on bootstrapped datasets randomly sampled with replacement from the original. A random forest also only considers splits among a number of randomly selected features in the dataset. The model finally makes predictions by averaging predictions across all decision trees. We used the Python library sci-kit learn as a basis for our random forest implementation.

We also generated feature importances in order to determine which features the model found most significant. In order to calculate the feature importance, first we calculate how much each feature contributes to decreasing the weighted variance of a tree during the training phase. The feature importance is calculated by averaging the decrease in variance over the trees. It is important to note that this approach is biased towards continuous features [12].

The top ten feature importances calculated for our model are displayed in Figure 3.4. Historical enrollment-related features displayed the highest scores, with past enrollment rated the most important feature by far, followed by sequencing. Title relevance was also rated relatively highly, indicating that our simple NLP representation seems to capture this feature well. Interestingly, course CUE guide scores were not given as much weight as department-wide CUE guide scores. This may indicate that CUE scores are similar within departments, and departments themselves have a larger impact on students' enrollment decisions. It is also possible that students may be more restricted by concentration requirements when taking departmental courses; CUE scores may be more relevant to the General Education courses we excluded from our final dataset.

Support Vector Machine

We also implemented a support vector machine as outlined in the CS96 report in order to compare our results to prior literature [2]. SVMs are supervised learning
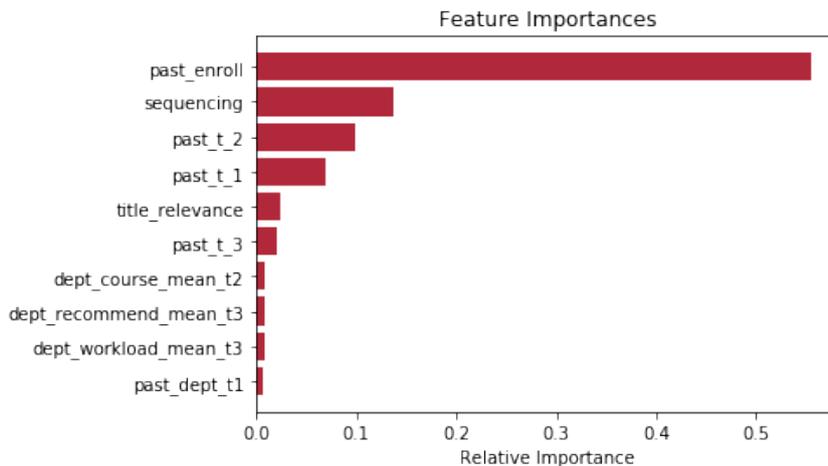
Figure 3.4: Ten most relevant features in our Continued Course Random Forest model by feature importance.

models that utilize kernel functions to map input data into high-dimensional feature spaces. See Smola and Schölkopf for more information on using SVMs for regression [22].

In our SVM implementation, we used a non-linear kernel function because the problem was too complex to model accurately with a simple linear function. Specifically, we selected a Gaussian kernel for its generalizability and to replicate the model parameters described by the CS96 report. We determined hyperparameters through a progressive grid search. We used the Python library sci-kit learn to develop our SVM model.

3.4.3 Per Student Models

One of the most unique datasets we had access to was per-student enrollment. Our per student models attempt to utilize this information and predict enrollment on a student-by-student basis, e.g. whether a student will take a certain course. In the prediction stage, the individual enrollment predictions are summed to derive an enrollment figure for each course.

Logistic Regression

A logistic regression uses the logit function to model a categorical dependent variable: $ln(\frac{P}{1-P}) =_0 +_1 X_1 +_2 X_2 ... +_n X_n$, where P is the probability that $Y = 1$ and $1 - P$ is the probability that $Y = 0$. The algorithm aims to create a linear deci-

sion boundary separating the classes $Y = 1$ and $Y = 0$ from each other based on the conditional probability of the class occurring dependent on previous data. Our model is a binary logistic regression that outputs a prediction for whether a student will take a course in a given term (1) or not (0). We selected the logistic regression because it reasonably fit the the binary nature of our problem and we wanted to compare the method outlined in the Johnson article [15]. Unlike the Johnson article, which used features such as major, the only student information we had access to was enrollment. We encoded this information as binary vectors indicating which courses students had taken prior to the predicted term.

Our model was trained on students who took courses in AY 2017-2018, one year before the test dataset. The model was trained separately for each predictor course. The training data was a matrix of binary vector encodings of courses taken prior to AY 17-18, with the test data being a binary indicator of whether the student had taken the predictor course in AY 2017-2018. The model was fit on the AY 17-18 data and a prediction was output for student data in AY 18-19. The final prediction was the sum of predicted outputs for a course, combining individual student enrollment predictions to an enrollment figure for the predictor course. This training and prediction process was repeated for each course in the dataset.

Simple Markov Model

Our simple Markov model extends the logic at the core of the logistic regression model. However, instead of fitting a model on student enrollment data, the model treats courses as states and generates transition probabilities between states based on student enrollment patterns. For example, consider the limited set of courses [COMPSCI-20, COMPSCI-50, COMPSCI-61]. Let's say 3 students who took COMPSCI-20 went on to take COMPSCI-50 the following semester and 2 went on to take COMPSCI-61. Then the transition probability from COMPSCI-20 to COMPSCI-50 is $\frac{3}{3+2} = 0.6$. The transition probability from COMPSCI-20 to COMPSCI-61 is $\frac{2}{3+2} = 0.4$. The model also calculates initial probabilities by taking the relative frequency of courses taken during a student's first semester. For example, say 4 students took COMPSCI-50 their first semester and 1 students took COMPSCI-61. Then the initial probability for COMPSCI-50 would be $\frac{4}{3+2} = 0.8$. This training phase is performed across all students and courses in the AY 17-18 dataset.

In the prediction phase, the model considers every student who has taken a

course offered in AY 18-19. For each of these students, the model sums the initial probabilities of the predictor course and the transition probabilities to the predictor course from all other courses the student took the term prior. Consider the case where 2 students took COMPSCI-20 in Spring 2018. Then for Fall 2018, the model would predict COMPSCI-50 to have an enrollment of $2 * 0.6 + 0.8 = 2$.

3.5   Models for Courses Offered Once

3.5.1   Reduced Feature Set

Given that courses offered for the first time do not have access to historical records, we had to reduce the feature set to limit the training data to relevant information. We disregarded historical course data such as past enrollment, but retained descriptive course data and historical department data.

3.5.2   Set Enrollment Models

Mirroring our preliminary models for the courses offered more than once, we implemented some static predictors for comparison. Our **Pred = 0 Model** simply predicts 0 for each enrollment figure in order to set a worst case standard. Our **Pred = Average** model calculates the average of the prior enrollment figure for courses offered more than once, and returns this number for every predicted course.

3.5.3   Clustering Models

Our clustering model makes predictions using the following steps. First, the model identifies a clustering rule for courses based on some combination of attributes. The clustering is then applied to courses that were offered more than once, and a prediction mapping is generated by averaging all past enrollments for courses in each cluster. The clustering is then applied to the test data, for which the predictions are generated from the mapping described previously.

Department

Our department clustering model created a simple clustering by course department. We reasoned that this would create a more accurate and specific prediction than simply an overall average.

Hierarchical Clustering

We attempted to create a more intelligent grouping through agglomerative hierarchical clustering. Hierarchical clustering is a clustering method that calculates the distance between each data point and groups the closest points together. Each observation starts in its own cluster and pairs of clusters that are closest together in distance are merged together. This allows us to generate a dendogram (Figure 3.5) which plots this merging process against the distance between each cluster. We can see from the dendogram that the optimal number of clusters is three, as it best balances grouping the closest points while differentiating between clusters.
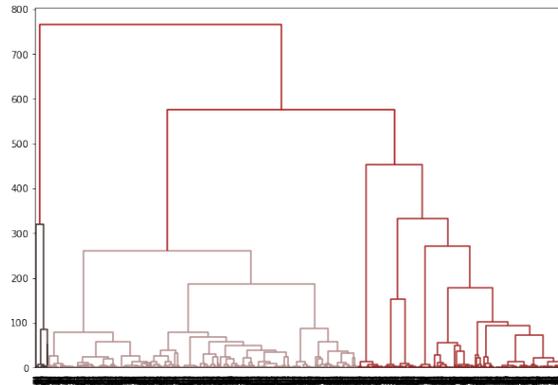
Figure 3.5: Dendogram describing the agglomerative hierarchical clustering of courses, using the Ward's minimum variance method. The clustering was performed on the reduced feature set of the dataset of courses offered more than once.

3.5.4 Retrained Models

Finally, we retrained our **Random Forest** and **SVM** models as described in the Continuous Course Models section on the reduced feature set.

The top ten feature importances calculated for our model are displayed in Figure 3.6. Title relevance had the highest score by far, indicating again that our NLP feature representing topic relevance seems to capture student preference very well. General Education status also had a relatively high importance where it did not for continued courses. This makes sense because General Education status most likely does not drastically change student enrollment in courses students already know about, whereas new courses are more appealing if they can knock off a General Education requirement as well.
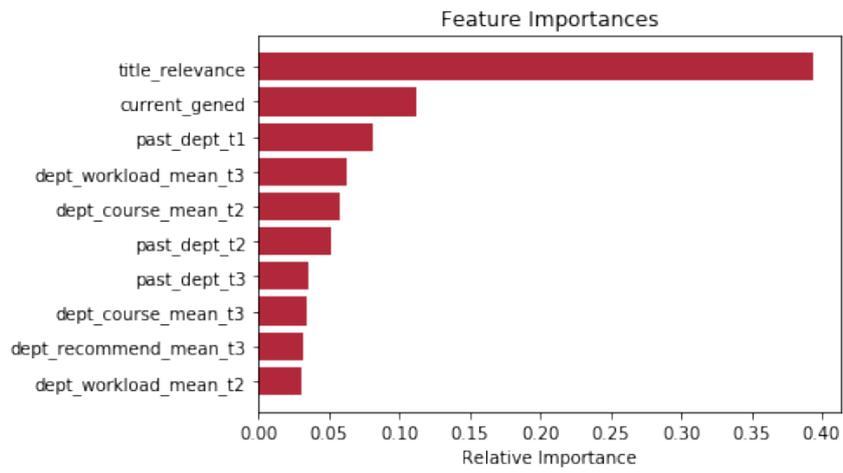
Figure 3.6: Ten most relevant features in our New Course Random Forest model by feature importance.

Results

4.1 Evaluation

4.1.1 Baselines

In order to evaluate our models against existing prediction methods, we defined some baseline metrics.

Departmental Predictions

The first step in the current course enrollment prediction process is the preliminary departmental predictions. We were able to use predicted enrollment, section size, and number of sections as a baseline with the prediction data we received from the Office of Undergraduate Education. We limited our data to the subset for which departmental predictions were available.

The departmental prediction process varies significantly between departments and neither the process nor the individual ultimately inputting the prediction is formalized. However, from qualitative interviews we determined departments seem to use factors such as past enrollment and topic relevance in order to generate these predictions, although not in a standardized manner. Some departments have more information than we have access to, such as instructor appeal and word of mouth. However, this prediction process is the one we are attempting to model using features as proxies for human reasoning, and makes for the closest baseline to beat.

OUE Predictions

The second baseline we used was the OUE preliminary enrollment predictions. These predictions are generated by the OUE considering the same factors as the depart-

ment heads as well as the departmental predictions. Given that the OUE has access to these departmental predictions, is able to ask heads about the reasoning behind these predictions, and may have their own outside information following trends in other courses, we should expect this baseline to be at least as accurate as the departmental predictions.

OUE Studycard Allocation Baseline

The OUE study card allocations are made after students submit their study cards to register their courses. The only changes that can be made in course enrollment after this deadline are add/drops. Thus there should be less variance from these allocations and this baseline should be the most accurate. This allocation is important because it is our best proxy for the best performance that can be expected from instituting a pre-registration system. Realistically, a pre-registration system should perform worse because students do not have the advantage of the information they now gain through shopping week, information that might impact whether they add/drop a course. We include this baseline as a soft upper limit to the degree of accuracy we can expect a model to achieve.

4.1.2 Methods

In our evaluation, we separate out courses offered in AY 2018-2019 as a test set. We extract features from historical data available before this year.

A significant concern for our machine learning models was overfitting. It is possible that the model becomes overly specific to the training data, and does not generalize well to future predictions. In order to avoid inaccurately evaluating our models, we implemented k-fold cross validation.

In k-fold cross validation, the dataset is split into k subsets, one of which is held out as test data. The model is trained on the remaining $k-1$ subsets and used to make predictions for the test data. The process is repeated with each of the k subsets held out as test data. We used the commonly accepted parameter $k = 10$.

4.1.3 Metrics

Given the unique nature of Harvard's teaching fellow assignment system, we had to develop specific evaluation metrics to capture different aspects of the problem and

rate each model. Two of these metrics, Expected Teaching Fellow Error and Expected Section Size Deviation, were derived from the CS96 report [2]. However, they were modified to incorporate the new information that section size is not standard across all departments or even all courses. We also included two additional metrics to capture raw enrollment prediction power and measure the number of new TFs who would be expected to "split fifths." We used the four metrics outlined below for evaluation.

1. **Expected Student Error (ESE):** Let y be the actual enrollment number for a course and $\hat{y}$ be the prediction. Then ESE is calculated $|\hat{y} - y|$. For example, if we predict an enrollment of 20 for a course which actually had 25 students, the ESE would be $|20 - 25| = 5$. The average, median and standard deviation of accumulated ESEs are calculated for comparison. The ESE attempts to capture the direct prediction accuracy of the model by calculating the average number of students our model deviates by. While this is not the most relatively useful metric for our problem, it is a useful initial evaluation of our models.

2. **Expected Teaching Fellow Error (ETFE):** Where ESE captures the direct prediction accuracy of our model, ETFE attempts to capture an aspect of prediction relevancy. ETFE is the difference between the predicted number of teaching fellows for a course and the actual number of section slots available, given that teaching fellows are assigned to sections in a 1:1 ratio. The number of teaching fellows is calculated TFs = max(1, round(enrollment / section size)). ETFE is then calculated $|TF(\hat{y}) - TF(y)|$. For instance, if we predict an enrollment of 20 for a course which actually had 25 students and which had a final section size of 5, the ETFE is calculated $|4 - 5| = 1$. This metric is directly useful in determining how many TF would need to be reassigned given a model prediction.

   ETFE is adapted from the CS96 model metric; where ideal section size was assumed to 15, we use available OUE final section size data instead. We define two versions of ETFE. The first only calculates average, median, and standard deviation for data where the final section size is available. While this is an accurate measure, it is not able to evaluate every prediction and results may be skewed. The second version, ETFE-15, addresses this by imputing a section size of 15 following the SEAS standard instead.

3. **Expected Section Size Deviation (ESSD):** ESSD is another prediction

relevancy metric that targets an area not covered by the first two metrics. If the predicted enrollment or number of TFs is too small for a larger course, students can be reasonably distributed across sections. However, for a smaller course, TFs may need to be hired or fired or reassigned. The absolute difference in predicted and actual enrollment is less relevant in this case.

ESSD instead calculates the section size deviation. To start, we calculate the section size that would result from maintaining the predicted number of TFs without regard to the actual enrollment. In the example outlined previously, we predicted 4 TFs for 20 students. If the actual enrollment were 25 and we distributed these students among the predicted TFs, we would have a section size of 6.25. Since the final section size is 5, this results in an ESSD of 1.25.

As before, ESSD and an imputed ESSD-15 were calculated. An average, median, standard deviation of ESSDs were returned.

4. **Expected TF Parity Deviation:** Another metric some departments use to evaluate their enrollment predictions is change in TF parity. In some departments, TFs are required to teach two sections. In the ideal scenario, TFs can teach both sections for one course. However, if courses require odd numbers of sections, some TFs may have to "split fifths," or teach one section in each of two different courses. The difference in workload and energy between the two scenarios is significant. Thus if a model predicts an even number of sections for a course but the actual number turns out to be odd, the cost is significant for any TFs that may have to be assigned. ETFPD captures the percentage of courses for which final section size is available where the predicted number of sections is even but the actual number is odd. This metric is relevant but essentially random, as none of our models focus on parity. However, it is a significant component in Harvard's section allocation system and without it our set of evaluation metrics would not be complete.

4.2 Quantitative Results

4.2.1 Continued Courses

The AY 18-19 dataset of continued courses (courses offered more than once) consisted of 1,920 entries. 1,046 of these entries were associated with departmental

and OUE prediction data, and 917 entries were associated with final section size numbers. For our departmental prediction, OUE prediction, and study card baselines, we only included the 1,046 entries that had this information in our test data. We calculated ESE, ETFE-15, and ESSD-15 across the entire test dataset; ETFE, ESSD, and ETFPD were calculated only with the 917 entries that had associated final section sizes.

Our final results are tabulated in Figure 4.1. We found that the study card baseline performed the best, as expected, as it takes into account course enrollments once students' course schedules are finalized. In particular, the predicted number of teaching fellows were off by an average of 0.3 while the predicted section size deviated by 1.8 students on average. The expected teaching fellow parity deviation was also very low, at only 6.8% of courses deviating to an odd number of teaching fellows from a predicted even figure.

Model	ESE	ETFE	ETFE-15	ESSD	ESSD-15	ETFPD
Dept Pred	12.075	1.273	1.128	5.407	5.681	18.8 %
	18.095	2.666	2.529	8.650	8.366	
OUE Pred	11.515	1.130	0.996	5.497	5.716	17.3 %
	17.626	2.522	2.367	9.816	9.593	
OUE Alloc	4.126	0.302	0.266	1.833	2.300	6.8%
	8.426	1.013	0.950	4.237	4.669	
Past Enroll	7.353	1.078	0.581	4.466	3.656	18.0 %
	14.146	2.471	1.756	7.500	6.200	
Sequencing	8.484	1.208	0.648	5.080	4.111	15.5 %
	18.11	3.023	2.138	9.253	7.404	
Random Forest	7.470	1.069	0.573	4.397	3.643	17.8 %
	15.554	2.374	1.686	7.970	6.476	
SVM	8.285	1.121	0.603	5.422	4.169	17.4 %
	22.003	2.660	1.891	15.111	11.126	
Logistic	17.898	1.567	0.860	23.686	15.931	2.3%
	35.323	3.472	2.578	40.114	30.198	
Markov	12.002	1.191	0.665	11.213	8.275	9.45%
	25.182	2.675	1.985	19.125	14.734	

Figure 4.1: Table of results for existing course models. Each model's results are recorded in two rows. The first row contains the average error for each metric, except for ETFPD, which is returned as a percentage. The second row records the standard deviation for each metric.

Taking a look at the distribution of errors across our baselines, we can see a similar trend in shape (Figure 4.2). In general, departmental and OUE enrollment predictions seem to be distributed symmetrically across the actual figure, while ETFE is skewed left and ESSD is skewed right. This indicates that there are more outliers where the preliminary predictions underestimate the number of TFs.

Interestingly, the study card allocations give an ESE distribution that is skewed right. This indicates that more courses see a significant number of students dropping the course rather than adding the course.
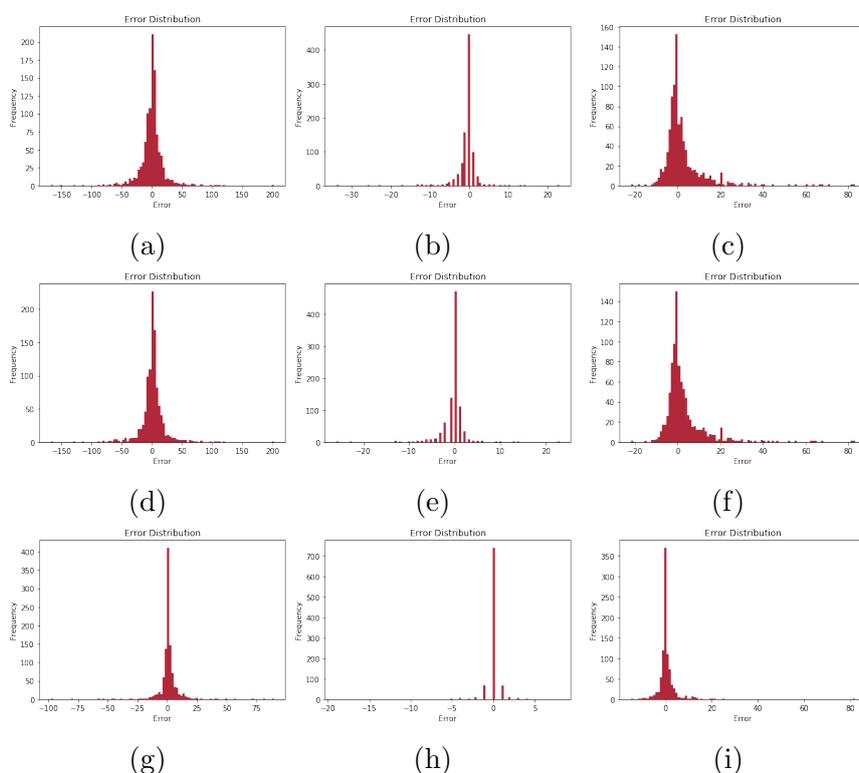
Figure 4.2: Distribution of errors for human baselines, calculated for existing courses.

Analyzing our model implementations, we see the the lowest average error was achieved by our Random Forest across all metrics other than ESE. Our top three systems by performance were Random Forest, Past Enrollment, and SVM.

All three models perform better than the departmental and OUE prediction baselines. The Random Forest model results in an ETFE of 1.069, Past Enroll in 1.078, and SVM in 1.121. When the ETFE-15 is calculated over the complete dataset, the error decreases. The Random Forest model results in an ETFE-15 of 0.573, Past Enroll in 0.581, and SVM in 0.603.

We can compare the significance of these results by applying it to a simplified TF allocation process for AY 18-19. In this example, we use the ETFE-15 metric in order to capture a larger sample of the data. Assuming that the OUE prediction taken over 1,046 courses for which this data was available, is a good approximation for the remaining 874 FAS courses offered that year, the baseline ETFE-15 is 0.996. This means that for the total 1,920 courses offered in AY 18-19, there would have been around 1,912 misallocated sections. If we assume that each teaching fellow is responsible for an average of 1.5 sections, this translates to 1,274 teaching fellows who have to be hired or fired after shopping week. If the enrollments returned by our Random Forest model had been used instead, there would only have been 1,100 misallocated sections translating to only 733 affected TFS. This is an improvement of 541 TFs over the human baseline. The OUE study card allocation returned an ETFE-15 of 0.266, translating to 510 misallocated sections and 340 TFs even after the study card deadline.

Our human baseline performs significantly better than the one used in the CS96 paper, which makes sense because the CS96 human baseline was calculated from a much smaller sample according to survey results. [2]. The CS96 machine learning model also returned a smaller ETFE of 0.361. This may be due to several different factors. The CS96 ETFE was calculated over all courses, not excluding new courses as we did. New courses are likelier to only be offered once or in the short term, and as we saw in Section 2, enrollment in courses offered once is significantly smaller on average. Combining this with the standard use of 15 as section size in the CS96 paper, including new courses in the dataset would decrease the ETFE significantly. Finally, the CS96 model incorporated some different features than our model due to data availability. However, we do not place as much weight on this last reason because our features overlapped with the CS96 model for the most part and we don't expect the excluded features to have an overly significant effect.

It is worth noting that the median ETFE and ETFE-15 across our Past Enrollment, Random Forest, and SVM models was 0. 72% of predictions by our Random Forest model returned an ETFE-15 of 0. 89% of predictions were within 1 TF, and 94% were within 2.

Using ETFE, we can see the cost of fixing TF misallocation; using ESSD we can examine the cost of ignoring it. We see that among our predictor systems, the Random Forest also returned the lowest average ESSD of 4.397 and ESSD-15 of 3.643. The ESSD of the OUE prediction was 5.497 while the ESSD-15 was 5.716.

47

Given an ideal section size of 15 for a course, using the OUE prediction would return a post-study card section ranging in size from 9.3-20.7. Using the Random Forest model would return a section ranging in size from 11.4-18.6, a more reasonable range. The OUE allocation returned an ESSD of 1.833 and an ESSD-15 of 2.3, or a range in section size from 13.2-16.8. Given that the section size deviated this much after the study card deadline, we see that the Random Forest range of 11.4-18.6 is relatively good. We also note that the CS96 machine learning model returned a lower ESSD, which may be due to the reasons we outlined before.

Analyzing the results from the rest of our models, we see that our Past Enroll predictor performed better across the board than our Sequencing model, indicating that past enrollment is the best single feature predictor. Our Sequencing predictor only performed marginally worse, however, which is to be expected because it is derived from past enrollment.

Our per-student models performed the worst out of our machine learning systems. This makes sense, as our per student models were trained on less information than our per course models. It is also intuitive that the problem of predicting what courses every single specific student will take in a given semester would return a significantly larger margin of error than predicting the enrollment for a given course. However, both models had a relatively low ETFE, which seems to indicate that their predictions deviated the most for courses with large sections. Indeed, we found that the Logistic Regression model most mispredicted COMPSCI-50, ECON-10A, and ECON-10B, while the Markov model most mispredicted ECON-10A, LIFESCI-1B, and ECON-1152, all large courses. Given their large course and section size, the higher ESE has a relatively lower impact on the ETFE. Additionally, nearly all of the most mispredicted courses are introductory level (ECON-1152 is the only standout, although it could be considered introductory in topic). This follows intuitively from the fact that introductory courses do not have significant predictor courses, and enrollment must be inferred from the initial probability distribution. In addition to low ETFE, the Markov model in particular achieves performance close to or better than the human baselines in each metric other than ESSD and ESSD-15. This indicates that these per-student models are a promising area to approach with a larger student feature set.

We see from the above results that our Random Forest model demonstrates a significant improvement over the departmental and OUE predictions. Our SVM model, which we constructed to be most similar to the CS96 model, also shows a

significant improvement over the departmental and OUE prediction baselines.

Interestingly, our Random Forest only shows a marginal improvement over the Past Enroll model. This shows that past enrollment is the single most significant feature in predicting course enrollment. However, our Past Enroll model is directly comparable to the automatic baseline from the CS96 paper. The only difference is that the CS96 automatic baseline calculates prediction errors over all courses, imputing a mean of 26 for new courses. The CS96 results showed a significant improvement with the SVM model over the automatic baseline. This difference in results may also be attributed to the inclusion of new courses and the set ideal section size of 15. We discovered in Section 2 that the average enrollment in courses offered once was 11.14. Assuming that this is a more accurate representation of enrollment in new courses, this means the CS96 automatic baseline on average overpredicts the enrollment in new courses by 15, or one TF. Indeed, for new courses, the average ETFE returned by the automatic baseline was 1.122, very close to our overprediction estimate. This justifies the need for a separate model trained for new courses.

4.2.2 New Courses

The AY 18-19 dataset of courses offered multiple times consisted of 568 entries. 277 of these entries were associated with departmental and OUE prediction data, and 220 entries were associated with final section size numbers. For our departmental prediction, OUE prediction, and study card baselines, we only included the 277 entries that had this information in our test data. We calculated ESE, ETFE-15, and ESSD-15 across the entire test dataset; ETFE, ESSD, and ETFPD were calculated only with the 220 entries that had associated final section sizes.

Our final results are tabulated in Figure 4.3. We found that the study card baseline performed the best, as expected. In particular, the predicted number of teaching fellows were off by an average of 0.186 while the predicted section size deviated by 3.5 students on average. The expected teaching fellow parity deviation was also very low, at only 4.5% of courses deviating to an odd number of teaching fellows from a predicted even figure.

Taking a look at the distribution of errors across our baselines, we can see a similar trend in shape (Figure 4.4). In general, departmental and OUE enrollment predictions seem to be distributed symmetrically across the actual figure, while ETFE is skewed left and ESSD is skewed right, more extremely than for existing courses. This indicates that there are more outliers where the preliminary predic-

Model	ESE	ETFE	ETFE-15	ESSD	ESSD-15	ETFPD
Dept Pred	11.635	1.05	0.844	8.240	8.326	13.6 %
	13.896	2.981	2.667	13.205	12.333	
OUE Pred	11.397	0.940	0.750	9.396	9.140	13.2 %
	13.956	2.873	2.564	14.195	13.003	
OUE Alloc	4.581	0.186	0.148	3.582	4.090	4.5%
	7.326	0.663	0.591	8.086	7.607	
Pred=0	12.919	0.740	0.318	19.418	12.919	0 %
	19.426	1.801	1.214	28.150	19.426	
Pred=Avg	14.270	2.640	1.054	8.440	11.984	29.5 %
	15.989	5.036	3.288	25.927	18.933	
Dept Clustering	8.209	1.227	0.508	9.870	7.344	9.5%
	17.945	2.677	1.702	27.112	18.812	
Hierarchical Clustering	8.233	1.363	0.559	10.379	7.573	15.4 %
	17.435	2.841	1.802	27.126	18.869	
Random Forest	8.438	1.013	0.471	8.559	6.215	16.8 %
	19.060	2.266	1.474	17.632	12.481	
SVM	7.422	1.086	0.452	11.121	7.129	7.7%
	17.920	2.379	1.516	27.521	18.975	

Figure 4.3: Table of results for new course models. Each model's results are recorded in two rows. The first row contains the average error for each metric, except for ETFPD, which is returned as a percentage. The second row records the standard deviation for each metric.

tions underestimate the number of TFs.

The study card allocations also give an ESE distribution that is skewed right. This indicates that more courses see a significant number of students dropping the course rather than adding the course.

Analyzing our model implementations, we see that among our machine learning models, the lowest average error was achieved by our Random Forest across all metrics other than ESE and ETFE-15. Our SVM model performed the best in these two metrics. However, the Random Forest achieved comparable errors for these two metrics and significantly smaller variance in ESSD and ESSD-15, which led us to label it our top-performing model. Our clustering models performed similarly, and only moderately worse than our Random Forest and SVM models.

Our Random Forest outperformed the departmental prediction baseline in ESE,
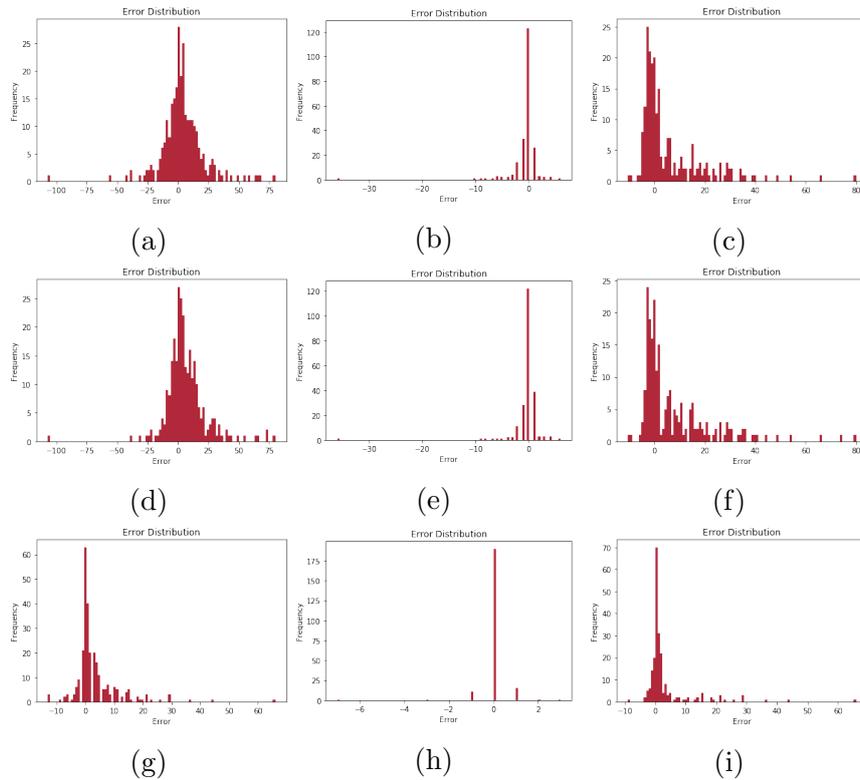
Figure 4.4: Distribution of errors for human baselines, calculated for new courses.

ETFE, ETFE-15, and ESSD-15, and outperformed the OUE prediction baseline in ESE, ETFE-15, ESSD and ESSD-15. However, we found that ETFE-15 and ESSD-15 might be less informative for new courses. New courses generally had an average enrollment of 11 so 15 might not be representative of an ideal section size. Thus, we could not decisively conclude that the Random Forest model could outperform human baselines. However, the Random Forest model and each of our other machine learning and clustering models all achieved a significantly lower ESE than the human baselines. This indicates that these models better predict the exact enrollment, but are not able to optimize for section size due to lack of data. These results show that further research into modeling new courses is a promising avenue, especially looking into acquiring and incorporating data such as the average OUE section size and prediction for new courses.

Interestingly, our Pred=0 model outperformed both the departmental and OUE prediction baselines. This indicates that new courses required very few TFs, and the departmental and OUE predictions often overestimate this figure.

51

Discussion

5.1 Impact and Implications

In this paper, we outlined the build and evaluation of a system to predict course enrollment.

For existing courses, we found that our Random Forest model had the best performance. The Random Forest model performed marginally better than the Past Enrollment model and modestly better than the SVM model. We also show that past enrollment is the most significant predictor feature based on two results: (1) the high-level performance of the Past Enrollment model and (2) the high Random Forest feature importance score. All three models performed significantly better than our human baselines, the departmental and OUE predictions.

For new courses, we found that the Random Forest model again performed the best of our machine learning models. The Random Forest model performed marginally better in varying metrics than the departmental and OUE predictions, but showed no significant improvements. This may be because new courses rely heavily on outside information and reasoning not represented in our feature set, such as knowledge of concentration requirements and instructor appeal. The Pred=0 model performed surprisingly well, especially for the ETFE metric, relative to the human baselines. This indicates that departments generally overestimate the number of TFs necessary for a course.

Across both types of courses, we found that the Random Forest model assigned high feature importance scores to title relevance. This indicates that (1) topic relevance is a significant predictor of enrollment and (2) our simple NLP feature based on course title is surprisingly good at capturing this topic relevance. No other course enrollment model in the literature has used natural language processing to capture qualitative aspects of courses - this suggests that incorporating NLP-based

features into course enrollment prediction is a promising avenue of further research.

Based on our results, we believe that the adoption of a Random Forest model in course enrollment predictions specifically for existing courses would significantly improve accuracy and provide a helpful guideline for departments and the Office of Undergraduate Education when determining sectioning. A machine learning model should not replace human baselines entirely. There are areas that still benefit from human review and need to be tweaked subjectively based on the situation, i.e. the ETFPD. Another finding was the significance of past enrollment in prediction. The Past Enrollment model performed nearly as well as the Random Forest model. Standardizing the departmental prediction process to incorporate this feature more heavily should improve accuracy further.

On the other hand, we do not currently see strong evidence in support of a machine learning model for new course enrollment predictions. None of our models are able to outperform the human baselines across all evaluation metrics. However, all of our models are able to better predict the actual enrollment than the human baselines. This supports further research into a future machine learning model specifically for new courses. Additionally, qualitative features, e.g. schedule-related information and instructor ratings, are intuitively more important considerations for new courses on which students have little to no information. Access to this data and the incorporation of these features would help to further improve the accuracy of our new course models.

Finally, we find that the current human baselines overestimate enrollment figures for new courses; incorporating this information in future departmental and OUE predictions should further improve accuracy.

5.2 Limitations and Future Work

Our model is limited by the data available. Given our dataset, we were unable to incorporate features outlined in previous work such as instructor appeal and schedule-related information. We were also limited to the AY 18-19 term as the test dataset because this was the period for which we obtained sectioning data.

In the future, we propose extending the system outlined in this thesis to be useful in implementation. In order to be ready for practical use, the system should standardize enrollment data sources and automate the extraction of features.

We also propose further research into the problem of enrollment prediction for

new courses. For the reasons outlined in the previous section, we believe that incorporating more qualitative features should improve our models.

Given the positive results supporting our title relevance feature, we suggest additional study into using NLP in course enrollment prediction. NLP seems to be a promising way to capture qualitative attributes students use to evaluate their course options. The use of more advanced NLP models could more accurately represent features like topic relevance.

Finally, we propose further study into the topic of TF parity, defined specifically for this thesis. Currently, none of our models optimize for this metric and human intervention is necessary to minimize ETFPD. Future research could identify whether there are good ways to incorporate TF parity into models, or whether this aspect should be left to humans.

Appendix

A.1   Glossary

A succinct glossary of terms related to course enrollment at Harvard College.

Committee on Course Registration: The faculty committee formed in 2019 to review current course enrollment procedures. The committee aims to come to a decision regarding course enrollment, including on issues such as shopping week and preregistration, in two years.

Course Lottery: A process by which some courses with limited enrollment select students, e.g. application or random algorithm. In particular, Expository Writing and General Education hold lotteries for all courses in the departments.

General Education: A set of requirements each College student is required to fulfill. Prior to Fall 2019 there were eight specific General Education subjects. Beginning Fall 2019, a General Education department encompassed these subjects. Some courses in other departments can also be used to fulfill General Education requirements. Used to be referred to as Core Curriculum.

my.harvard: The online registration system through which students can enroll in courses. The transition to my.harvard from physical study cards occurred in 2014.

Preregistration: A proposed course registration system under which students would enroll in courses prior to the start of the semester.

Shopping Period: The first week of an academic semester at Harvard before students are required to formally enroll in courses. Students may use shopping period to sit in on courses and review assignments and syllabi before submitting their study cards.

Study Card: Course enrollment forms which students are required to submit at the end of shopping period. Beginning in 2014, study cards are submitted online on my.harvard.

Teaching Fellow (TF): TFs at Harvard may lead sections, grade assignments, supervise projects, and/or take on other duties related to the teaching of a course.

A.2   Implementation Details

Raw data was cleaned and stored as CourseInfo objects as defined in the skeleton code below:

```python
class CourseInfo:
    def __init__(self, name, term, enrollment, title,
                 instructor_ids, dept, huids, gened):
        pass

    # getter functions
    def get_inorder(self, d):
        pass

    def get_enrollment(self, term):
        pass

    def get_huids(self, term):
        pass

    def get_instructor(self, term):
        pass

    def get_instructor_change(self):
```

```python
        pass

    def instructor_changed_from_last(self):
        pass

    def first_time_offered(self):
        pass

    def get_gened(self, term):
        pass

    def gened_changed_from_last(self):
        pass

    def current_gened(self):
        pass

    def get_terms(self):
        pass

    def get_enrollment_inorder(self):
        pass

    def get_course_mean(self, term):
        pass

    def get_recommend_mean(self, term):
        pass

    def get_workload_mean(self, term):
        pass

    def get_dept_est(self, var):
        return self.dept_est[var]
```

```python
    def get_oue_prelim_est(self, var):
        pass


    def get_oue_studycard_alloc(self, var):
        pass


    def get_final_alloc(self, var):
        pass


    # setter functions
    def set_name(self, name):
        pass


    def set_title(self, title):
        pass


    def set_dept(self, dept):
        pass


    def set_term(self, term):
        pass


    def set_enrollment(self, term, enrollment):
        pass


    def set_huids(self, term, huids):
        pass


    def set_instructor(self, term, instructor_id):
        pass


    def set_gened(self, term, gened):
        pass


    def set_cue(self, term, course_mean,
```

```
                    recommend_mean, workload_mean):
        pass


    def set_dept_est(self, enrollment,
                    target_section_sz, num_sections):
        pass


    def set_oue_prelim_est(self, enrollment,
                        target_section_sz, num_sections):
        pass


    def set_oue_studycard_alloc(self, enrollment,
                        target_section_sz, num_sections):
        pass


    def set_final_alloc(self, enrollment,
                    avg_section_sz, num_sections):
        pass


    def append_huids(self, huids, term):
        pass


    # constrain dataset to certain terms
    def delete_term_on(self, term):
        pass
```

Student data was cleaned and stored as StudentInfo objects as defined in the skeleton code below:

```
class StudentInfo:
    def __init__(self, huid):
        pass
    # getter functions
    def get_terms(self):
        pass
    def get_courses(self, term):
```

```
        pass
    # setter functions
    def add_course(self, term, course_id):
        pass
    # return whether student took course_id
    def took_course(self, course_id):
        pass
    # return all courses taken
    def courses_taken(self):
        pass
```

The current code repository cannot be made public because it contains sensitive data. However, scrubbed code will be made available upon request.

A.3 Data Cleaning Methodology

Specific decisions made in the data cleaning process are outlined below:

The course identifier was stored as SUBJECT-NUMBER, all uppercase, no special characters.

The following subjects were mapped in order to establish consistency:
- AI, AESTHINTP → AESTHINT
- CULTRBLF → CULTBLF
- EMREASON → EMREAS
- ETHREASON → ETHRSON
- IMUNOL → IMUIL
- SCILIVSYS → SCILIVSY
- SCIPHYUNV → SCIPHUNV
- AMSTUDIES → AMSTDIES
- CLASSTDY → CLSSTDY
- HINDIURDU → HINDURD
- MODMIDEAST → MODMDEST

Additionally the following adjustments were made:
- ECON-1123A1 was combined into ECON1123
- ECON-1010a1, ECON-1010a2 → ECON-1010A
- COMPSCI-50 (Letter-Grade), COMPSCI-50 (SAT/UNSAT) → COMPSCI-50

Departments were standardized according to the following code:

```python
def standard_dept(dept):
    # combine engineering depts
    eng = ['APPLIEDCOMPUTATION',
           'APPLIEDMATHEMATICS',
           'APPLIEDPHYSICS',
           'COMPUTERSCIENCE',
           'ENVIRONMENTALSCIENGINEER']
    # standardize dept formatting
    dept = dept.upper()
    dept = dept.replace('AND', '&')
    dept = re.sub('[\W_]+', '', dept)
    # map departments for consistency
    if 'CLASSICS' in dept:
        return 'CLASSICS'
    elif 'WOMEN' in dept:
        return 'WOMENGENDERSEXUALITY'
    elif 'AFRICAN' in dept:
        return 'AFRICANAMERICANSTUDIES'
    elif 'RELIGION' in dept:
        return 'RELIGION'
    elif 'ENGINEERING' in dept or dept in eng:
        return 'ENGINEERINGAPPLIEDSCIENCES'
    elif 'EASTASIA' in dept:
        return 'EASTASIANSTUDIES'
    elif 'ARCHL' in dept:
        return 'ARCHITECTURELSCAPEARCHITECTUREURBANPLANNING'
    elif 'ENGLISHAMERICANLIT' in dept:
        return 'ENGLISHAMERICANLITERATURELANGUAGE'
    elif 'GERMANICLANG' in dept:
        return 'GERMANICLANGUAGESLITERATURES'
    elif 'RUSSIAEEUROPE' in dept:
        return 'RUSSIAEASTERNEUROPECENTRALASIA'
    elif 'STEMCELL' in dept:
        return 'STEMCELLREGENERATIVEBIOLOGY'
```

```
elif 'DRAMATIC' in dept: #combine dramatic arts and tdm
    return 'THEATERDANCEMEDIA'
elif 'ORGANISMIC' in dept:
    return 'ORGANISMICEVOLUTIONARYBIOLOGY'
elif 'ENVISCIENCE' in dept:
    return 'ENVIRONMENTALSCIENCEPUBLICPOLICY'
elif 'BIOSCIENCES' in dept:
    return 'BIOLOGICALSCIENCESINPUBLICHEALTH'
elif 'ARTFILMVISUALSTUDIES' in dept:
    return 'VISUALENVIRONMENTALSTUDIES'
elif 'NEAREASTERNLANG' in dept:
    return 'NEAREASTERNLANGUAGESCIVILIZATIONS'
elif 'ROMANCELANGUAGES' in dept:
    return 'ROMANCELANGUAGESLITERATURES'
return dept
```

Bibliography

[1] K. R. Balachandran and D. Gerwin. Variable-work models for predicting course enrollments. *Operations Research*, 1973.

[2] D. Bemis et al. CS96 final report: An investigation into the early course selection issue. *Computer Science 96*, January 2004.

[3] J. Berger and M. McCafferty. Faculty council approves proposal to retain shopping week. *The Harvard Crimson*, March 2019.

[4] J. Berger and M. McCafferty. Shopping week to stay — for now. *The Harvard Crimson*, May 2019.

[5] M. Bernhard. Huit team charged with developing student information system seeks feedback. *The Harvard Crimson*, May 2014.

[6] R. Britney. Forecasting educational enrollments: Comparison of a markov chain and circuitless flow network model. *Socio-Economic Planning Sciences*, September 1974.

[7] X. Chen and H. Ishwaran. Random forests for genomic data analysis. *Genomics*, 99(6):323 – 329, 2012.

[8] Rodriguez-Galiano et al. Machine learning predictive models for mineral prospectivity: An evaluation of neural networks, random forest, regression trees and support vector machines. *Ore Geology Reviews*, 71, 2012.

[9] Faculty Committee on Course Registration. Report of the committee on course registration. March 2019.

[10] Faculty of Arts and Sciences. Course registration. `https://courseregistration.fas.harvard.edu/`. [Online; accessed 2.21.2020].

[11] A. Fu and L. Wang. Faculty talk elimination of shopping week. *The Harvard Crimson*, March 2018.

[12] U. Grömping. Variable importance assessment in regression: Linear regression versus random forest. *The American Statistician*, January 2012.

[13] Harvard FAS Registrar's Office. CUE Guide - Harvard CUE. `https://q.fas.harvard.edu/`. [Online; accessed 2.21.2020].

[14] D. P. Hopkins and W. F. Massy. *Planning Models for Colleges and Universities.* Stanford University Press., 1981.

[15] B. Johnson and S. Strohkorb. Predicting course enrollment. *Frankly Speaking*, April 2014.

[16] M. Kaletzky. Harvard institute of technology? *The Harvard Crimson*, June 2007.

[17] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *Forest*, 23, 11 2001.

[18] F. Murtagh and P. Contreras. Algorithms for hierarchical clustering: an overview. *WIREs Data Mining and Knowledge Discovery*, December 2011.

[19] R. O'Brien. Kirby tables preregistration proposal. *The Harvard Crimson*, March 2003.

[20] F. et al. Ouallouche. Improvement of rainfall estimation from msg data using random forests classification and regression. *Atmospheric Research*, October 2018.

[21] M. Serrano-Megías and J. López-Nicolás. Application of agglomerative hierarchical clustering to identify consumer tomato preferences: influence of physicochemical and sensory characteristics on consumer responsen. *Journal of the Science of Food and Agriculture*, December 2005.

[22] A. Smola and B. Schölkopf. A tutorial on support vector regression. *Statistics and Computing*, August 2004.

[23] Undergraduate Council Education Committee. Shopping week survey findings. Fall 2018.