Course Suggestions and Insights on edX

Harika Malapaka
Student of Comp. Science
North Carolina State University
Raleigh, NC, USA
(001)984-218-6410
hsmalapa@ncsu.edu

Apurva Dilip Bakshi
Student of Comp. Science
North Carolina State University
Raleigh, NC, USA
(001)908-992-5321
abakshi@ncsu.edu

ABSTRACT
Educational data mining (EDM) is a branch of data mining and machine learning research to develop new ways to analysis educational data from an educational system. The work in this paper is to develop a system suggesting courses along with other relevant insights like gender distribution in courses and countries on (with respect to) edX platform. This system is designed to ease a student who is new to edX and wants to pick a course based on his interest. This module would ask his topics of interest and display a course which matches his topic of interest along with that course importance levels (how important is it to do that course), grade distribution and number of students who dropped out of that course and if a student gives his details like his country, Year of birth and education level, it will display best course which his peers choose. This way it becomes easy for the new student to make decisions for picking (auditing or registering) a new course.

Keywords
Data Mining, MOOC, edX, Machine Learning, Feature Engineering, E-Learning, Decision Trees, Gaussian Naïve Bayes, Collaborative Filtering, User based Filtering, Python

1. INTRODUCTION
E-learning theory describes the cognitive science principles of effective multimedia learning using electronic educational technology. Cognitive research and theory suggest that the selection of appropriate concurrent multimedia modalities may enhance learning as shown in [1], as may application of several other principles.

1.1 Objective
There are many students on MOOC's who often register for a course but don't end up completing the course. This is a common problem almost in all E-learning platform. One of the reason for this problem may be because, the user before registering doesn't know the course background, what kind of people prefer this course, or how challenging is this course, or what age group of people prefer this and so on. So, this system presents a module to suggest a best course for a user new to edX. This is already present in most of the Universities, so adding this feature to a popular platform like edX would help people who prefer studying online. He can view different charts showing the distribution of courses with respect to age, gender, country, educational level of student etc. Given his basic details like topic of interest, year of birth, highest educational level and country, the system will present him with 2 types of courses: one is as per his topic of interest, showing how many certified, how many dropped out and grade distribution of that course. The other type of suggestion is course which many of his peers took (like-minded people).

1.2 edX and its Data
edX is a Massive Open Online Course (MOOC) provider. It hosts online University level courses in a wide range of disciplines to a world-wide student body including some courses at no charge. It also conducts research into learning based on how people use edX platform. edX differs from other MOOC providers such as Coursera and Udacity, in that it is a non-profit organization and runs on free open edX Open source software platform. The Massachusetts and Harvard University created edX in May 2012. More than 70 schools, nonprofit organizations offer courses on edX.

edX consists of weekly learning sequences. Each sequence is composed of short videos interspaced with interactive learning exercises where students can easily and immediately practice the concepts they learned by watching videos .The courses often include tutorial videos that are similar to small on campus discussion groups. There is an online discussion forum where people can review questions and comments to each other and teaching assistants. edX offers certificated of successful completion and some courses are credit eligible.

1.3 Comparing with Other's Work
As proved in the paper [2], The research on edX data is vast. Many works can be related to this paper. Few of the below mentioned works are from research done at Harvard University.

While Harvard is truly re-imagining what is possible on-campus, there is a conscious effort to preserve the residential Harvard experience.

By using the online platform and residential classrooms to advance generalizable knowledge, Harvard is able to learn more about student learning and use the research findings to enhance the classroom educational experience.

1.3.1 Similar work in Drop-outs
6 MA students from Harvard University made a research in paper [3] and [4] on why do students drop out. The point of argument was regarding their country (i.e their background).

Often, the most underserved learners are those in less economically developed nations [5], where the native language is not English. However, because English is the primary language for MOOCs, for learners with beginning and intermediate English language ability, authentic materials are often beyond their language proficiency and may become incomprehensible without help. This is a weak point where students tend to drop the course.

1.3.2 Adaptive learning featured in Harvard X course

The correspondent in Harvard found a system that features adaptive learning and assessment algorithms that tailor course material in response to student performance. In paper [6], The broader mission is to make sure that students are really benefiting from the online learning experience.

Adaptive learning programs are very good at speeding up information acquisition and lengthening retention, as well as individualizing learning to help learners see where they have difficulty.

1.3.3 How Grades are affected?

In 2013-14, Harvard University piloted the use of MOOCs as tools for blended learning in select undergraduate and graduate residential and online courses [7]. One of these courses, The Ancient Greek Hero, combined for-credit (Harvard College and Harvard Extension School) and open online (HarvardX) groups into a single online unit, marking the first time the same instance of a MOOC was used simultaneously by both tuition-paying, credit-seeking students and non-paying, non-credit students enrolled exclusively online. In this research, they analyze and compare the online behavior of students and participants in the three groups that simultaneously participated in The Ancient Greek Hero via the edX platform.

It was found that, in similar fashion to a traditional learning setting, students enrolled in all three versions of the course engaged the online content in a transactional way, spending their time and effort on activities and exercises in ways that would optimize their desired outcomes. While user behavior was diverse, HarvardX participant engagement tended to be either very deep or virtually nonexistent, while College and Extension School students displayed relatively homogenous patterns of participation, viewing most of the content but interacting mostly with that which affected their overall course grades.

Ultimately, it was concluded that educators who intend to utilize MOOC content in an effort to apply blended learning techniques to their classrooms should carefully consider how best to incorporate each online element into their overall pedagogical strategy, including how to incentivize interaction with those elements.

2. PROBLEM STATEMENT

All the above works were researched on why student drop out, or study patterns of students who are online (e-Learners versus offline) i.e students attitude towards grade in a classroom.

In general, recommendation can be user-based or content-based approaches [8] or both (i.e hybrid approach). In the user-based approach, the users' preferences are analyzed and aggregated from the users' profile. In the content-based approach, the system finds patterns or similar patterns related to previous experiences in order to recommend new ones. The later approach is a form of case-based reasoning approach. This system covers both the approaches for building the suggestion system.

The system in this paper covers all these individual segments and builds a course suggestion module. Student behavior in MOOC is considered in this system. The main weak point is, student will read/learn as per his interest, but may not give the exam at the end.

Before registering for a course, the student can see the metrics about that course, so that he will get a better knowledge to make a decision on the course.

A user can decide before hand the quality of the course, or how interesting is it, by observing the number of drop outs.

He can find out the certificate seekers of a particular course. What kind of courses did his like-minded people take up.

Collaborative filtering helps us to know what people with similar interest have done. In this system, we make use of CF (collaborative filtering) in selecting a course for a new student.

3. BACKGROUND

3.1 Data Mining

Data mining is the computing process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems. It is an essential process where intelligent methods are applied to extract data patterns. Data mining is the analysis step of the "knowledge discovery in databases" process, or KDD

The knowledge discovery in databases (KDD) process as described in paper [9]is commonly defined with the stages:

(1) Selection
(2) Pre-processing
(3) Transformation
(4) *Data mining*
(5) Interpretation/evaluation.

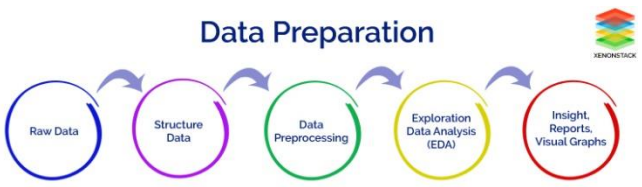

Figure 1: Data Preparation Steps

Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from different sources it is collected in raw format which is not feasible for the analysis.

Therefore, certain steps are executed to convert the data into a small clean data set. This technique is performed before the execution of Iterative Analysis. The set of steps is known as Data Preprocessing. It includes Data Cleaning, Data Integration, Data Transformation and Data Reduction.

3.2 Machine Learning

Machine learning [10] is a method of data analysis that automates analytical model building. It is a branch of artificial intelligence based on the idea that machines should be able to learn and adapt through experience.

Because of new computing technologies, machine learning today is not like machine learning of the past. It was born from pattern recognition and the theory that computers can learn without being

programmed to perform specific tasks; researchers interested in artificial intelligence wanted to see if computers could learn from data. The iterative aspect of machine learning is important because as models are exposed to new data, they are able to independently adapt. They learn from previous computations to produce reliable, repeatable decisions and results. It's a science that's not new – but one that has gained fresh momentum.

4. DATASET

HarvardX and MITx released deidentified dataset [11] of their student's activity in 2014 academic year. Data from the first year (Academic Year 2013: Fall 2012, Spring 2013, and Summer 2013) of MITx and HarvardX courses on the edX platform

The 20 attributes of the dataset are:

- course_id: administrative, string, identifies institution (HarvardX or MITx), course name, and semester, e.g. "HarvardX/CB22x/2013_Spring"
- userid_DI: administrative, string, first portion identifies dataset (MHxPC13 corresponds to MITx HarvardX Person-Course AY13), second portion is a random ID number. Example ID: "MHxPC130442623".
- registered: administrative, 0/1; registered for course, =1 for all records in person-course.
- viewed: administrative, 0/1; anyone who accessed the 'Courseware' tab (the home of the videos, problem sets, and exams) within the edX platform for the course.
- explored: administrative, 0/1; anyone who accessed at least half of the chapters in the courseware (chapters are the highest level on the "courseware" menu housing course content).
- certified: administrative, 0/1; anyone who earned a certificate. Certificates are based on course grades, and depending on the course, the cutoff for a certificate varies from 50% -- 80%.
- final_cc_cname_DI: mix of administrative (computed from IP address) and user-provided (filled in from student address if available when IP was indeterminate); during de-identification, some country names were replaced with the corresponding continent/region name. Examples: "Other South Asia" or "Russian Federation".
- LoE: user-provided, highest level of education completed. Possible values: "Less than Secondary," "Secondary," "Bachelor's," "Master's," and "Doctorate."
- YoB: user-provided, year of birth. Example: "1980".
- gender: user-provided. Possible values: m (male), f (female) and o (other).
- grade: administrative, final grade in the course, ranges from 0 to 1. Example: "0.87".
- start_time_DI: administrative, date of course registration. Example: "12/19/12".
- last_event_DI: administrative, date of last interaction with course, blank if no interactions beyond registration. Example "11/17/13".
- nevents: administrative, number of interactions with the course, recorded in the tracking logs; blank if no interactions beyond registration. Example: "502".
- ndays_act: administrative, number of unique days student interacted with course. Example: "16".
- nplay_video: administrative, number of play video events within the course. Example: "52".
- nchapters: administrative, number of chapters (within the Courseware) with which the student interacted. Example: "12".
- nforum_posts: administrative, number of posts to the Discussion Forum. Example: "8".
- roles: administrative, identifies staff and instructors, but blank as staff and instructors were removed from this release.
- inconsistent_flag: administrative, identifies records that are internally inconsistent. Due to the two different sources, if something is wrong with the Tracking Logs for a class or a student, then records in Person Course can be internally inconsistent and have a value of '1' in this column.

Table 1: The list of courses in the dataset

Institution	Course Code	Short Title	Full Title	Semester
HarvardX	CB22x	HeroesX	The Ancient Greek Hero	Spring-Summer 2013
HarvardX	CS50x	-	Introduction to Computer Science I	Fall 2012 – Spring 2013
HarvardX	ER22x	JusticeX	Justice	Spring-Summer 2013
HarvardX	PH207x	HealthStat	Health in Numbers: Quantitative Methods in Clinical & Public Health Research	Fall 2012
HarvardX	PH278x	HealthEnv	Human Health and Global Environmental Change	Summer 2013
MITx	14.73x	Poverty	The Challenges of Global Poverty	Spring 2013
MITx	2.01x	Structures	Elements of Structures	Spring-Summer 2013
MITx	3.091x	SSChem	Introduction to Solid State Chemistry	Offered twice: Fall 2012 and Spring 2013
MITx	6.002x	Circuits	Circuits and Electronics	Offered twice: Fall 2012 and Spring 2013
MITx	6.00x	CS	Introduction to Computer Science and Programming	Offered twice: Fall 2012 and Spring 2013
MITx	7.00x	Biology	Introduction to Biology – The Secret of Life	Spring 2013
MITx	8.02x	E&M	Electricity and Magnetism	Spring 2013
MITx	8.MReV	MechRev	Mechanics Review	Summer 2013

Table 2: Number of Students in each course

Course	Number of students
HeroesX	30,002
IntrotoCS	1,69,621
JusticeX	57,406
HealthStat	41,592
HealthEnv	39,602
Poverty	27,870
Structures	5,665
SSChem	14,215+6139
Circuits	40,811+22,235
CS	66,731+57,715
Biology	21,009
E&M	31,048
MechRev	9,477

The courses Solid State Chemistry, Circuits and CS were offered twice (i.e during Spring and Fall). So the students were added in their corresponding rows in the column 'Number of Students'. We can see even in table 1 that these courses are offered twice.

5. Classifiers Used for Machine Learning

When the data are being used to predict a category, supervised learning is also called classification. This is the case when assigning an image as a picture of either a 'cat' or a 'dog'. When there are only two choices, it's called two-class or binomial classification. When there are more categories, as when predicting the winner of the NCAA March Madness tournament, this problem is known as multi-class classification**.**

5.1 Decision Tree

Tree has many analogies in real life, and turns out that it has influenced a wide area of machine learning, covering both classification and regression [12]. In decision analysis, a decision tree can be used to visually and explicitly represent decisions and decision making. As the name goes, it uses a tree-like model of decisions. Though a commonly used tool in data mining for deriving a strategy to reach a particular goal, it's also widely used in machine learning, which will be the main focus of this article.

A decision tree is drawn upside down with its root at the top. In the image on the left, the bold text in black represents a condition/internal node, based on which the tree splits into branches/ edges**.** The end of the branch that doesn't split anymore is the decision/leaf**,** in this case.

5.2 Gaussian Naïve Bayes

In a classification problem, our hypothesis (h) may be the class to assign for a new data instance (d). One of the easiest ways of selecting the most probable hypothesis given the data that we have that we can use as our prior knowledge about the problem. Bayes' Theorem provides a way that we can calculate the probability of a hypothesis given our prior knowledge.

Bayes' Theorem is stated as:

$$P(h|d) = (P(d|h) * P(h)) / P(d)$$

Where:

P(h|d) is the probability of hypothesis h given the data d. This is called the posterior probability.

P(d|h) is the probability of data d given that the hypothesis h was true.

P(h) is the probability of hypothesis h being true (regardless of the data). This is called the prior probability of h.

P(d) is the probability of the data (regardless of the hypothesis).

Naive Bayes can be extended [13] to real-valued attributes, most commonly by assuming a Gaussian distribution.

This extension of naive Bayes is called Gaussian Naive Bayes. Other functions can be used to estimate the distribution of the data, but the Gaussian (or Normal distribution) is the easiest to work with because you only need to estimate the mean and the standard deviation from your training data.

6. METHOD

The task for this system was implemented using Python Programming language. PyCharm, an IDE of Python was used.

The value of data is enhanced by visualization. The Django Web framework is a good tool for this purpose, as well as to create complete Web apps in Python. Thus the UI was built using Django. This interactive suggestion system can be broken down into the following functions as detailed below.

6.1 Data Collection and Cleaning

The data from the data verse (Harvard.edu) was collected and imported to PyCharm as a Data Frame. The package 'Pandas' was used for this purpose.

The cleaning process included removing inconsistent rows which are rows with '1' as value in the column "incomplete_flag". The unnecessary columns for this project like 'roles', 'user-id', 'event start time', 'event end time' are removed. Then even the column 'incomplete_flag' itself was removed.

There are many courses which are offered in spring and fall. They are mentioned separately in the dataset using a name like HarvardX/CB22x/Spring_2015 and HarvardX/CB22x/Fall_2016. We combined both into a single course using course name from the course table, i.e 'The Ancient Greek Hero' in this case. While combining the course data, we have assumed that the course material/structure have not changed and both the courses are the same.

There were many rows where grade was '0' even though student attempted quizzes, watched threshold number of videos, made forum posts. So, considering that he has some knowledge, his

grade can be predicted (a different value from 0) by applying machine learning. Decision Tree Classifier was used for this purpose taking maximum depth as Decision Tree and maximum leaf nodes as 4.

The training data for this classifier is the value of columns (no of days active, no of forum posts, no of videos played, no of chapters read) where the value in the column grade is not '0' and the labels are grades. The testing data is same as training data with different rows where value of the column grade is '0'.

Results of data cleaning are as follows:

Before:

- Total records **: 6,41,139**
- Number of columns : **20**
- Number of students: **4,76,532**
- Number of courses: **16**

After:

- Total records: **5,40,977**
- Number of columns : **15**
- Number of students: **4,17,482**
- Number of courses: **13**

6.2 Statistics on Best course:

6.2.1 Certified

To determine the importance of a course, the function 'certified', returns the percentage of people who got certified. This function filters students who certified and then are grouped based on course. So for a suggested course, the percentage of people who are certified is given along with the course name.

The number of people who are certified may be a wrong notion because if there are very few people taking up the course, there are few people who will be certified. So, the percentage of people who are certified is given rather than the number of people who are certified.

6.2.2 Grade distribution

If the user wishes to use the grade in this course somewhere, i.e if he's interested about the grade, he can analyze the grade distribution beforehand for the suggested course. For the best course it displays the number of people who got an 'A' grade and so on in that course. We assumed the split of grades as follows:

- 0 to 0.2 --> D
- 0.2 to 0.5 --> C
- 0.5 to 0.8 --> B
- 0.8 to 1 --> A

As mentioned in the above section of Data Collection and Cleaning, the grades are first updated using Decision tree algorithm. Then they are grouped based on courses. When a course is suggested, the grades of that course are then organized in a data structure of grade followed by its occurrence. This table is presented to the user along with course name.

6.2.3 Dropouts

To see the quality of the course, or to determine the people who leave the course in the middle, the function 'dropouts' determines the number of people who have dropped out of the course.

A student can be tagged as drop-out based on his activity data. We have assumed that a student is dropped out if he hasn't registered, hasn't viewed at least half of study materials or haven't got a certificate, grade is '0' and the number of forum posts, videos played, number of days active and chapters read is '0'.

A new column is appended to the dataset representing the drop-out value. It will be '1' if the student drops out and '0' otherwise. This value is determined from the above mentioned attributes. Grouping this column by course name, we can calculate the number of people who have dropped out in a particular course.

6.3 Best course as per peers

This system prompts the new user to give basic details through a User Interface.

When a user enters his basic details like year of birth, his country of residence, gender and highest level of education, he will be suggested a best course which his peers (similar background people) selected. This is done through running a machine learning algorithm –Naïve Bayes (which is mentioned above).

Training data for this algorithm is value of columns 'gender', 'Year of Birth', 'highest level of education' and 'country' and the target/label/outcome variable is 'course name'. The split of training and testing data to measure accuracy of the classifier is 70% and 30%. When the user enters the value, they are passed to a 'predict' function, which will predict the best course.

The values passed to the classifier are not structured well because few attribute values are numbers and few are string. So, the input is pre-processed using label encoding and then they are trained. The label encoder converts all values to integers.

Similarly, when the classifier receives the 4 input values from the user, they are first pre-processed and then sent for testing to give the output which is a course name.

7. RESULTS:

7.1 Classifiers performance

For getting accurate grades, we have used Gaussian naïve Bayes because it proved to perform better than other algorithms like SVM which was taking lot of time, Regression because it's results were not that diverse. To find the best course as per their peers, we have used Decision trees because it's accuracy is high compared to Naive Bayes, SVM and Regression.

Accuracy of Gaussian bayes for predicting grades is '0.0' This is infact a good sign because we no more have zeros as grades for students who have done some learning.

The accuracy of Decision tree classifier for predicting best course is '0.67'. The maximum leaves is 4 and maximum depth is 3.

7.2 User Interface

Figure 2 shows the user interface where a student enters his basic details and his interests, and in return, he gets:

- Course suggested according to his topic of interest,
- A little description about that course
- Percentage of people who certified in that course
- Number of dropouts in that course
- Grade distribution of that course after re calculation of grades in all records where grade was 'zero'.
- The course which similar people selected depending on his basic details given by him

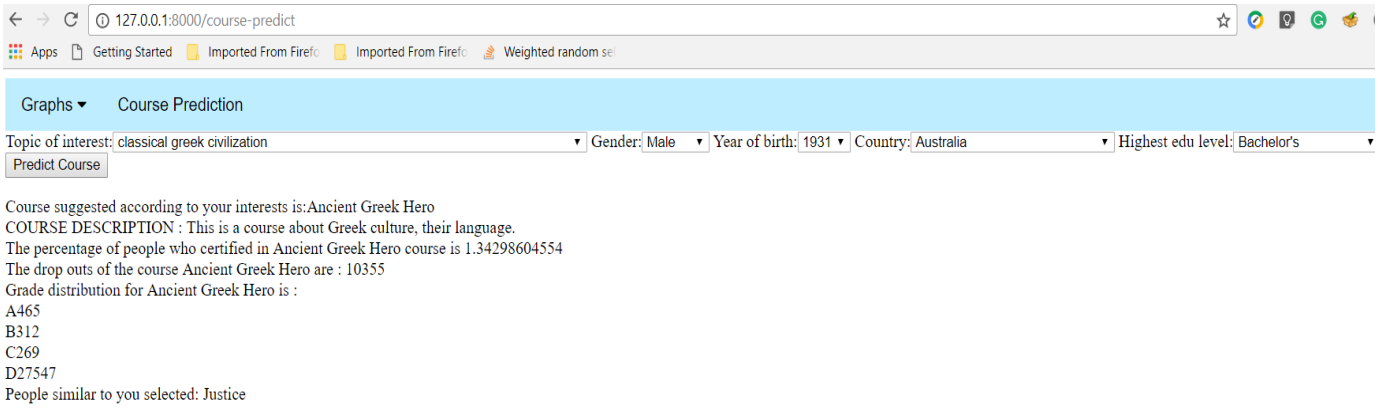

Figure 2: User Interface for course prediction system

7.2 Data Visualization

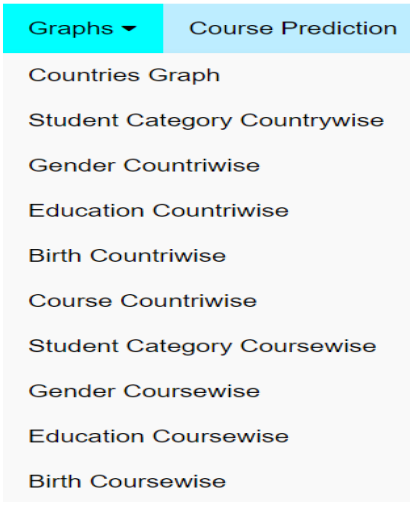

Figure 3: options available for visualizing data

We also have the options as shown in Figure 3 to visualize the student data so that user can see the distribution countrywise and coursewise and make some deductions in an area he is interested in. eg. If a student wants to know how many people of his country or gender or education take a particular course or if a student is interested in a particular course, he can details of that course, like age range, success rate etc.

These graphs can be used by a student to get an overview of courses as well as can be used by the MOOC to improvize their courses or make some decisions.

The graph in Figure 4 can be useful to the MOOCs to find out which area to concentrate its efforts on so as to affect maximum number of students. From the graph in figure 5, we can get an idea of level of education in each of the countries. By combining this graph with the student density graph, we can decide where, how much and in what way to put the efforts in.

From the graph in Figure 6, a student can check what age group students register for the course he is interested in, or which courses do his age group student mostly take.

From the graph in Figure 7, MOOC can decide what should be the difficulty level for each course by finding out the education level of students taking that course.

Figure 4: a colour gradient graph displaying the number of students in each country

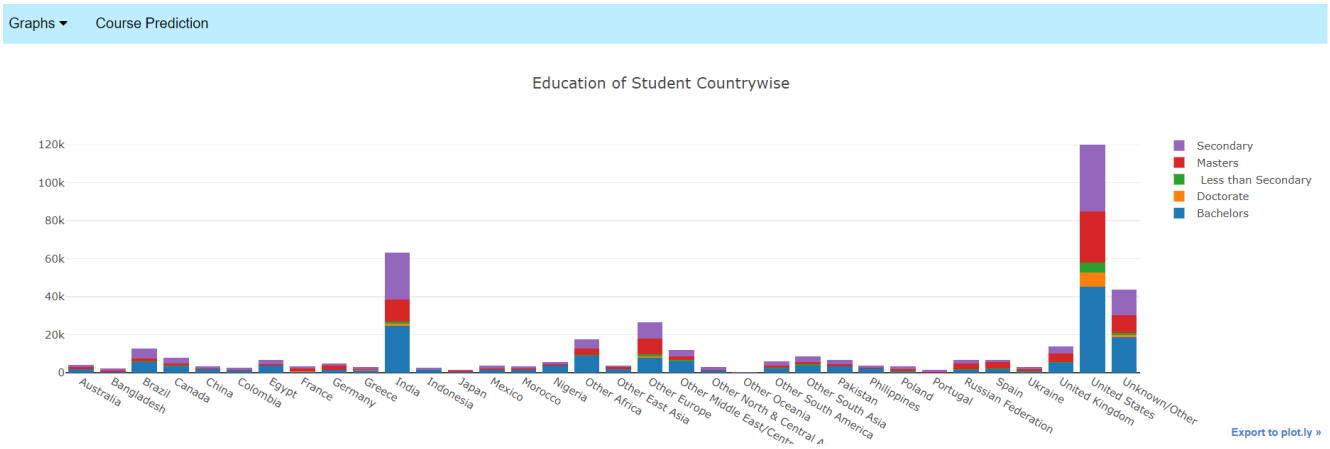

Figure 5: graph showing education of students countriwise

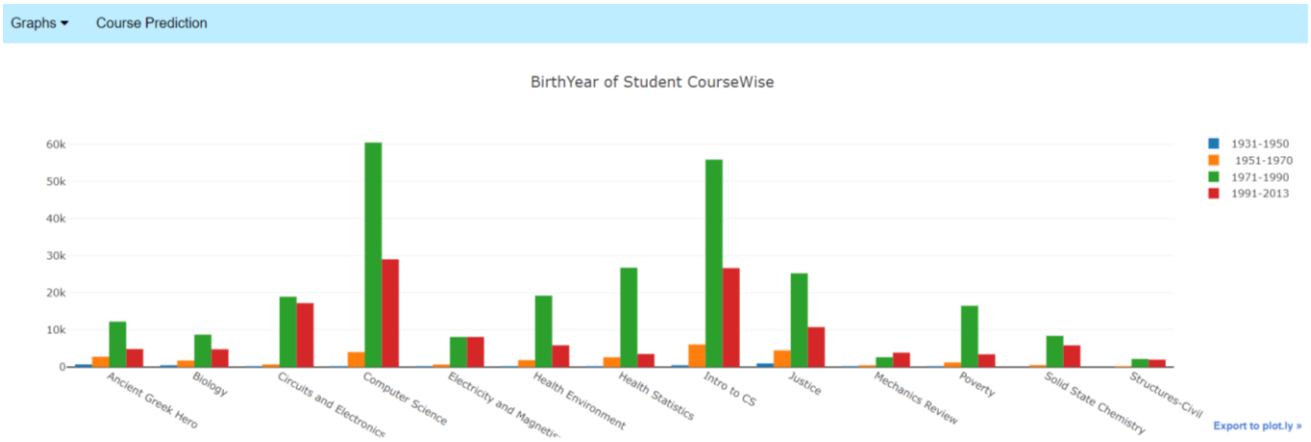

Figure 6: graph displaying registered student's age group for each of the course

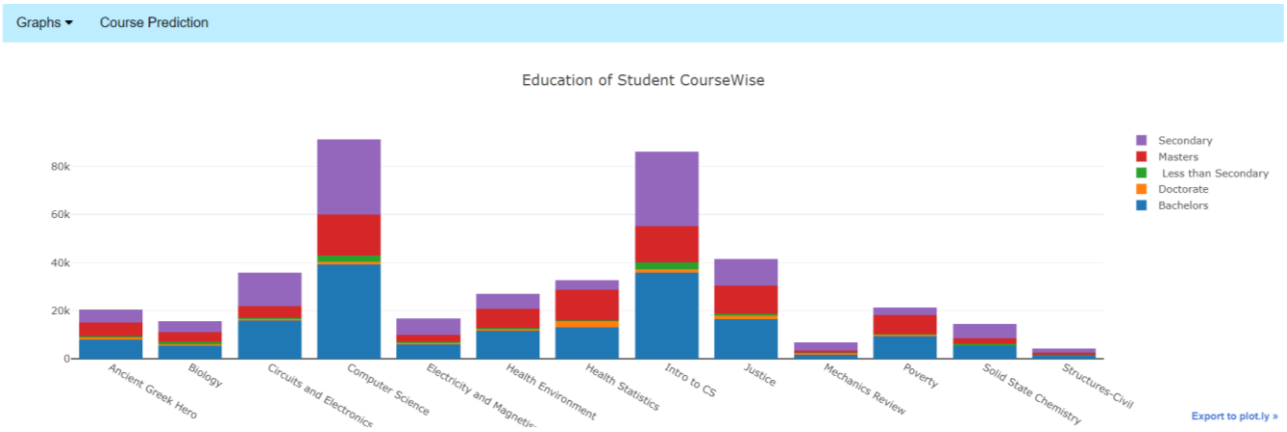

Figure 7: graph showing education level for each of courses

8. CONCLUSION:

The concept of selecting a good course before registering for it is already existing in many Universities. But many people these days are relying on online education. So it's good to have a UI which is collaborated with a MOOC platform which can provide some insights about courses with the existing (past) student and course data.

Student behavior in MOOC is considered in this system. We removed the notion that student with grade '0' didn't learn anything and calculated the new (approximate) grade based upon his overall course activity.

This system is built based on a dataset which accounts only small amount of MOOC data. Thus generalizations are based on this small amount of data. This can be considered as a caveat of this system.

9. ACKNOWLEDGMENT

The work in this paper is supported by North Carolina State University. We thank Dr. Collin Lynch for guiding us through-out this research. The conclusions, recommendations, findings etc found in this material are purely made by authors and do not reflect those of the North Carolina State University.

10. FUTURE WORK

- Making the UI more user friendly.
- Considering many topics of interests at an instance from user.
- Improving the accuracy of the algorithms.
- Integrating many other datasets so that we have many courses including many students

11. REFERENCES

[1] Prakash Kumar Udupi, Nisha Sharma, S K Jha, "Educational Data Mining and Big Data Framework for e-Learning Environment", 5th International Conference on Reliability, Infocom Technologies and Optimization (ICRITO) (Trends and Future Directions), Sep. 7-9, 2016.

[2] Harvard Publications https://vpal.harvard.edu/publications?page=1

[3] Whitehill, J., Mohan, K., Seaton, D., Rosen, Y., & Tingley, D.," MOOC Dropout Prediction: How to Measure Accuracy?.", Proceedings of the Fourth (2017) ACM Conference

[4] Whitehill, J., Williams, J. J., Lopez, G., Coleman, C., & Reich, J. (2015)," Beyond Prediction: First Steps Toward Automatic Intervention in MOOC Student Stopout.", Proceedings of the 8th International Conference on Educational Data Mining , 171-178.

[5] Turkay, S., Eidelman, H., Rosen, Y., Seaton, D., Lopez, G., & Whitehill, J., "Getting to know English language learners in MOOCs: Their motivations, behaviors and outcomes. ", Proceedings of the Fourth Annual ACM Conference, 2017

[6] Breslow, L., Pritchard, D. E., DeBoer, J., Stump, G. S., Ho, A. D., & Seaton, D. T.," Studying learning in the worldwide classroom: Research into edX's first MOOC" , 2013.

[7] Jeff Enamuel, Anne Lamb, "Open, Online and Blended: Transactional Interactions with MOOC.", 2015

[8] Richter, M. M., Weber, R.O., (2013). Case-Based Reasoning. Springer-Verlag Berlin Heidelberg.

[9] https://www.xenonstack.com/blog/data-preprocessing-data-wrangling-in-machine-learning-deep-learning

[10] https://www.sas.com/en_us/insights/analytics/machine-learning.html

[11] Source of Dataset

https://dataverse.harvard.edu/dataset.xhtml?persistentId=doi:10.7910/DVN/26147

[12] https://towardsdatascience.com/decision-trees-in-machine-learning-641b9c4e8052

[13] https://machinelearningmastery.com/naive-bayes-for-machine-learning/