The Student Carrier Guidance Recommendation System using Machine Learning

1. Mane Mrunal Suryakant, Computer dept, TSSM's PVPIT, Pune
2. Kaktikar Shivani Shivajirao, Computer dept, TSSM's PVPIT, Pune
3. Mohol Payal Rajaram, Computer dept, TSSM's PVPIT, Pune
4. Munde Shital Vasant, Computer dept, TSSM's PVPIT, Pune
5. Prof. N. S. Bagal ,Computer dept, TSSM's PVPIT, pune

Abstract

There has been exponential growth in the area of education, especially the education system has seen enormous growth, which has had some unforeseen impacts on the students. Due to the large increase in the choices, it has become increasingly difficult for the students to select appropriate courses. This leads to an increase in the number of dropouts and lower experience for the student. There have been several systems that have been developed that help the students choose a better course through their aptitude analyzed with the help f some tests. But it lacks a technique that can counsel students based on their emotional states, which would also account for their strengths and weaknesses. This paper proposes a hybrid recommendation model which utilizes the Hidden Markov Model and Collaborative Filtering to provide optimum recommendations to the students.

Keywords— *Hidden Markov Model, K Means clustering, Collaborative filtering, Hybrid recommendation***.**

I. INTRODUCTION

Recommendations are basically suggestions we come across from others in our day to day life. Actually, we all somehow depend on suggestions from others. For example, choosing a dress to wear we ask others for suggestions. Best recommender systems suggest best to the users. Recommender systems are extensively utilized in different areas, like in e-commerce. For example, if we are watching cartoon video then ads or recommendations should be related to kids. When we are talking about a student, recommendations are in the form of enhancement in academic areas or behavioral areas.

Do students need to understand where they stand? What are their strengths and weaknesses? In what areas they should improve? Whether they are good in academics? Are they punctual in their work? How is their behavior? What does their personality say? What can be the best domain for them? What is their mental ability and strength? These questions motivated me to select this topic for research and project work. Also, there may be a growing consciousness among researchers approximately the plain versions within the instructional overall performance of college students in tertiary establishments Machine learning techniques have been formulated as a paradigm in the modeling of student academic performance. Also, there are techniques which use recommender techniques for instructional data mining, especially for predicting student overall performance.

Educational institutions are more and more required to reveal performance in their students. This gives rise to a need to extract beneficial facts from the dataset of a scholar so that it will improve student retention rates. With the growing quantity of statistics on international wide internet and with significant upward push variety of customers, it becomes more and more essential for companies to go looking, map and provide them with the relevant chew of data consistent with their choices and tastes.

Optimization and scheduling is an important aspect of many processes from timing work shifts to assigning jobs to machines. As the number of professors and courses increase, students start considering different courses and teachers for their elective courses, and this problem draws the interest of researchers in the optimization field. Graduate or senior undergraduate students need to decide optional courses while preparing their course programs. There are many factors which affect the students' decision such as different times of courses, professor and course preferences, and conflicting hours of courses. These constraints become difficult to handle by hand or mind when the number of available courses reaches hundreds. In addition, students may not be aware of all the courses they can take. It also requires effort to find the best matching ones when there are many course alternatives.

In current years, there is an increasing interest in moving the testing platform from paper-based to online, and simultaneously, an interest in automating many of the tasks involved in conducting tests. One candidate for automation is the evaluation of assignments, which traditionally consumes more time. In recent times, there has been much research done on the automation of answer evaluation. Most of this work, however, is in relation to short answers (such as a phrase or a sentence). In tertiary institutions, online academic advising systems can provide prompt advice as and when required, and thus enhance the student experience and save staff time and other institutional resources. Therefore such systems are gaining popularity. Research into such systems and the development of such systems are in progress.

Clustering is a method of grouping records data into incoherent clusters so that the information inside the same cluster is comparable, however, facts belonging to special cluster vary. A cluster is a set of information item which can be common to each other are in the identical cluster and diverse to the objects are in other clusters. The call for organizing the pointy growing records and learning valuable facts from records, which makes clustering strategies are broadly carried out in many application areas including synthetic intelligence, biology, customer courting management, information compression, records mining, facts retrieval, photo processing, device mastering, marketing, remedy, sample recognition, psychology, records and so on. Cluster evaluation is a method this is used to find the traits of a cluster and to awareness on a particular cluster for in the additional analysis. Clustering is unsupervised learning and does not base on predefined instructions. In clustering, we measure the dissimilarity between data via measuring the gap between every pair of data. Those degrees encompass the Euclidean, distance.

The clustering algorithm is to classify a set of elements into several groups according to a certain rule; it appeared firstly in statistics, artificial intelligence and other fields; it is deeply studied by scholars. This paper uses the k-means clustering algorithm, one of the typical distance-based clustering algorithms, when two elements are closer, they are more similar, otherwise the opposite.

A statistical model, an i.e. difference on the Markov Chain. It is developed in the 1960s by L.E Baum and his teammate. It is the simplest form of Dynamic Bayesian Network. In the HMM model, there is an unobserved or hidden state in comparison to the standard Markov Chain Model in which all the states are visible. In extracting and machine learning filed this model is utilized for gesture and handwriting recognition, speech recognition, reinforcement learning, bioinformatics, etc. It is utilized to find the variable future state by using probabilities rely on the current and past state. The main dissimilarity between HMM and Markov Model is that in the previous case state is invisible directly but the output is visible.

The collaborative filtering recommendation is to use the past behavior of a user to analyze his interest preference to recommend his possible favorite items. The CF algorithms include the memory-depend and the model-depend. The memory-based looks for similar nodes through the nearest neighbor algorithm; it usually includes the user-based and item-based algorithm. The user-based argues that the scores of users who are most common to target user of item approximate target user's score of the identical item. The item-based holds that when predicting target user's score we can predict the target user' score of an item according to his scores of items which are most common to the identical item.

This research paper dedicates section 2 for analysis of past work as literature survey, section 3 deeply elaborates the proposed technique and whereas section 4 evaluates the performance of the system and finally section 5 concludes the paper with traces of future enhancement.

II. LITERATURE SURVEY

This section of the literature survey eventually reveals some facts based on thoughtful analysis of many authors work as follows.

G. Kaushik Ram, N. Sai Kiran and S. Sudha [1] present an optimal mix and develop a recommendation system to suggest similar products to users. A simple but effective optimization algorithm was developed to prioritize faster or cheaper delivery option as per the user's preference, which makes no allocation if the stock is insufficient. The end result was a mail sent to all users and suppliers which gives relevant data. The recommendation system algorithms resulted in suggesting other similar products that users might wish to purchase.

Hualong Ma, Xiande Wang, Jianfeng Hou, and Yunjun Lu [2], present semantic similarity analysis into course selection, realizes a course recommendation system. Course recommendation is worth researching. Using LSI + D2V to resolve the course recommendation can present the most reasonable result compared. Each course description is first modeled as a document, and a clean tokenize- TFIDF-Word2Vec-Doc2Vec pipeline is built to create vectors for each course from which cosine similarities will be calculated. The next step is to determine the number of courses they shall present, and the threshold of similarity according to real facts.

Suleyman Uslu, Can Ozturan and Mehmet Fatih Uslu [3], proposed an integer programming based solver to provide university students optimized course programs based on their ratings on courses and course timing conflicts. An online web platform, Mecanin, is built for the use of this optimum schedule tool by Boğaziçi University students. It is observed that this tool which is implemented in javascript is sufficiently responsive to suggest a schedule for students based on their course ratings and time conflicts of courses. Additionally, the recommendation algorithm depends on the CF method is proposed which considers the current courses of the user and the course preferences of other users which may be similar to the user in term of course programs. Various variants of this algorithm are also provided, and the results of these proposed algorithms and basic methods which are not based on similarity techniques are compared. It is observed that CF, which is a similarity-based method, outperforms all other methods. In addition, it is shown that by considering the conflicts, the score of the CF method could be improved.

Carolina Mejia, Sergio Gomez, Laura Mancera and Sibylle Taveneau [4], introduced an inclusive learner model for adaptive recommendation to favor and assist students with reading difficulties or dyslexia in virtual education. Future research may give a clear cut picture of the students and their differences, as well as, impact and satisfaction with recommendations. Thus, their next steps are center on research with past student logs in order to approve the model to find out the actions of new students; implementing inclusive components in new studying framework for virtual education, i.e., the tool for recover reading difficulties and cognitive deficits related to dyslexia; implementing the mechanisms to detect if students are capable of understanding and inspecting their own learner model through different visualization in order to generate awareness about their learning process; and integrating the components to adapt contents, activities, and tools, according to the learner model in the LMS.

Elham S.Khorasani, Zhao Zhenge and John Champaign [5], examined, using historical data from students studying computer science, the effectiveness of a Markov based collaborative filtering course enrollment recommended. They argued that the order in which courses are taken by students plays an important role in recommending new courses to students to take in their future semester. They showed that the precision and recall of the recommendations returned by the Markov model on this dataset outperforms those of item-based and matrix factorization-based recommender systems. They consider this work as an experimental study to test their early expectations about how to preprocess and analyze the enrollment data.

Moving forward they are concerned for examining finer grain recommendations for students. In their work, the precision of the recommendations is measured by taking the courses that students ultimately enrolled in as ground truth. Instead, a comparison of the recommendations made by a recommender system to recommendations made by experienced advisors is a worthwhile evaluation of the system that should be examined. Instead of predicting the courses that students actually enrolled in, they ultimately want to help students make better course selections than they would have made themselves.

They would like to compare students success, calculated by their GPA at graduation, to their fidelity to the recommended course of study. This fidelity can be computed by determining the ratio of courses taken to courses recommended. Their hypothesis is that students who chose a course of study closer to what would have been recommended will have greater success than students who did not. When data becomes available about student employment and salary after graduation, they are interested in examining correlations between transcript data and career success. One possible outcome would be to identify course results that are strongly predictive of long term success.

Anirudh Kashi, Sachin Shastri, Akshay R. Deshpande Jawahar Doreswamy and Gowri Srinivasa [6] proposed here a system that works well for answers with smaller amounts of variation from the expectation of the evaluators but needs more fine-tuning to process more application-oriented answers. They attempt to build an automatic evaluator for English answers larger than a single sentence. To do this, they will also gather and process results of manually evaluated tests as part of the project (the”dataset”). Given student submissions for a question, and the expected”correct” answer, marks will be recommended proportional to how similar the two are. It is to be noted that the scores their system produces are only recommendations, and not meant to be taken as correct always.

H. Slimani, N. El faddouli, R. Benslimane and S. Bennani Rime [7] presented the approach related to the personalization of search based on students’ interests and the recommendation based on their competences by modeling the student profile using decimal classification indices of Dewey. These indices are part of the metadata description of educational digital resources. They used this bridge in order to filter, recommend and organize educational resources during search operations to students on the graphical interface of ORI-OAI repository.

Muhammad Fahim Uddin, Soumita Banerjee, and Jeongkyu Lee [8] present a subset of overall research towards correlation personality features with academic and career data to improve success rates and decrease poor performances at schools and in the jobs. They show the application of their work towards building a novel framework for recommendation systems for an individual such as Bob who utilizes Alice as alumni personality relevance and output of PAE and his own choice to finally

get the recommendation score for his decision-making process. They show results for various parameters that support their work. Though, they seek to improve their results in the future to make it more linear as possible. Their work shows a potential of more research to utilize the unstructured data (social networking) and academic data through the lens of personality features to improve recommendation systems and create a personalized recommendation to select academics that has more success likeliness for individuals who seek recommendations.

David Simkins and Adrienne Decker [9], present the effects of the analysis of the open-ended questions at the cease of the survey. The students had been requested to discover a concept that they determined especially tough and why. They were also asked to explain how they finished fulfillment in getting to know an idea that they to start with finding hard. They developed a system of thematic codes using a grounded principle method on a sampling of the information or data, which they then carried out to the whole dataset. They uncovered styles and trends inside the student responses. At some point of their analysis, they have exposed that students are expressing blame for the dearth of learning on themselves and teachers, pointing to perceived failures within the classroom environment and course structure, and are showing proof of the expertise of the learning technique. They finish with a few typical tips for the way this locating may be operationalized to higher recognize the system for this intermediate learner.

Kathiravelu Ganeshan and Xiaosong Li [10], proposed an Internet-based system using collaborative filtering for advising intelligent pupil, this approach usually used in recommendation systems. This method assumes that users with common traits and behaviors could have comparable choices.

With their advising system, students are looked after into corporations and given advice considering the relevant elements and additionally considering their similarities to specific groups. A major use of the online advising system they have proposed and prototyped is to help students choose courses from over fifty courses and five interlinked pathways in the Bachelor of Computing (BCS) program. If a student belongs to a certain group, a course that other students in that group have preferred or performed well it may be recommended to the student. The system is developed to be integrated into their current

student management system, PeopleSoft. Therefore, their students don’t need to create a profile to use this system. Real student data with complete records for the last four years (2011 to 2014) of all 743 students enrolled in over 50 courses in the Bachelor of Computing Systems (BCS) was anonymized and used in training and testing the prototype. Data included academic transcripts as well as biographic data.

III PROPOSED METHODOLOGY

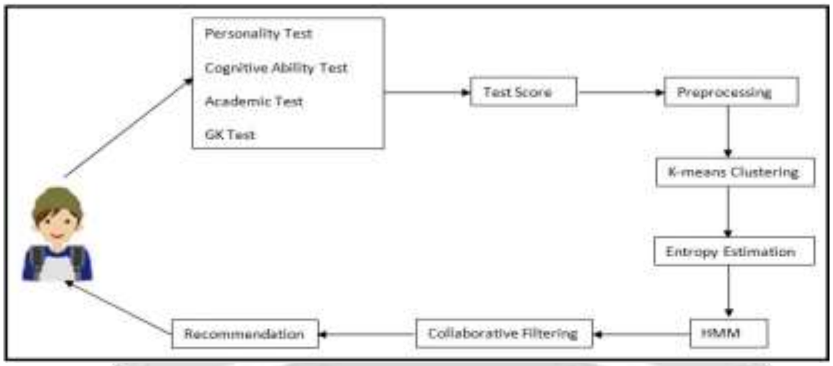


Figure 1: System Overview of the Proposed Model

The Proposed model Student carrier recommendataion is depicted in the figure . And the steps that are carried in the process of building of the system is narrated with the below mentioned steps.

Step 1: Student Data Collection – The Student is tested in four different areas of expertise, such as Cognitive Ability, General Knowledge, Academic and Personality tests through an interactive user interface.

The tests have been deployed with 100 questions pertaining to the various tests which are stored in the database. The questions are of a multiple-choice type and 10 random questions are retrieved and presented to the user during a test on a particular topic. For further calculations, the scores of the test are stored in the database once a student completes the tests.

Step 2: Pre-processing and Feature Extraction – After the completion of the tests, all the relevant scores and the data is retrieved from the database for further evaluation and clustering of the individual scores obtained in the various tests such as, Cognitive Ability test, General Knowledge test, Academic tests and Personality tests and are stored into a pre-processed list.

Step 3: Feature Clustering – The K-nearest Neighbour algorithm is utilized for the clustering of the pre-processed list. As the data has to be divided into rows, each row of the pre-processed list containing all the four tests is considered. Further, for every row contained in the pre-processed list, their corresponding Euclidean Distance is evaluated with respect to all the other rows. The Euclidean distance of the whole pre-processed list P_{ED} is obtained by calculating the average of all the Euclidean distances of all the rows.

The pre-processed list is then sorted by appending the corresponding Euclidean distance of each row sorting it ascendingly through these values. Centroids of the clusters need to be identified and are done so by randomly selecting the data points according to the required clusters. The centroids serve as a boundary for the cluster with the help of $P_{ED.}$ The clusters are then generated according to the policies of the K-nearest Neighbour algorithm.

Step 4: Entropy Estimation – The Entropy of the clusters is evaluated after their generation for every test of the student. To extract the Entropy, the current scores of the student are compared with the four scores of the test in the row. The Entropy of each cluster represents the importanance of the cluster based on the current test scores. So in this process cluster whose entropy is more than 0.5 is considered as the best one and it is selected for further use of Hidden Markov model. The estimation of the entropy can be estimated using the following equation 1.

$$E = -\frac{A}{C}\log\frac{A}{C} - \frac{B}{C}\log\frac{B}{C} \_\_\_\_\_\_\_ (1)$$

Where

C= 4 (As there are 4 tests are there)
A= matched score count
B= C-A
E = Entropy Gain

Step 4- Hidden Markov Model - The formed information gain clusters are used to find the Forward probability of the Hidden Markov model. The forward probability is estimated using the distance between the current score and each of the cluster's score. And the smallest distance cluster is considered in the next process of Baum Welch matrix Evaluation process. The forward Probability estimation can be shown in the below shown Algorithm1.

Algorithm 1: Forward Probability

```
// Input :  Current Score Set $C_{SET} = \{P_S , C_S, A_S, G_S \}$
[$P_S$ : Personality Test Score,  $C_S$ : Cognitive Test Score, $A_S$ : Academic Test Score ,  $G_{S\,:}$ General Knowledge Test Score ]
// $IG_{C\ :}$ Information Gain Cluster
// Output : Forward Probability Set $FP_{SET}$
**Function** : forwardProbability($C_{SET,}$ $IG_C$ )
Step 0: Start
Step 1: $MIN_{SCORE}$ =100
Step 2: for  i=0 to size of $IG_C$
    Step 3: $S_G = IG_{Ci}$
    Step 4: MEAN = ∅
    Step 5: for  j=0 to size of $S_G$
        Step 6:  ROW= $S_{Gj}$
        Step 7:D= $\sum ROW_k - C_{SETj}$
        Step 8:  MEAN=MEAN+D
    Step 9: End for
    Step 10: MEAN=MEAN/ $S_G$ Size
    Step 11: IF MEAN < $MIN_{SCORE}$
        Step 12:  $MIN_{SCORE}$ =MEAN
        Step 13: $FP_{SET\ =}\ S_G$
Step 14: End for
Step 15: return $FP_{SET}$
Step 16: Stop
```

In the next step of the Hidden Markov model this forward probability selected cluster is used for matrix translation Process. Here each current test score is searched in the respective columns from the forward probability selected cluster for their smallest distance row to yield a probability list row set of Baum Welch model.

Step 5- Colloborative Filtering - The Baum Welch set is used to estimate the best score from the probability set. Then this score is used to provide the Recommendation to the student based on the predefined suggestions stored in a workbook for the ranges of the obtained score of Baum Welch.

IV RESULTS AND DISCUSSIONS

The performance of the proposed system has been extensively evaluated and various experiments have been conducted on the test machine which is equipped with Core i3 Central Processing Unit paired with a physical memory of 4 GB. The System was implemented in a Java Environment on a NetBeans 8.0 IDE and the Database tasks were handled by the MySQL database server. Various tests to evaluate the accuracy and strength are elaborated below.

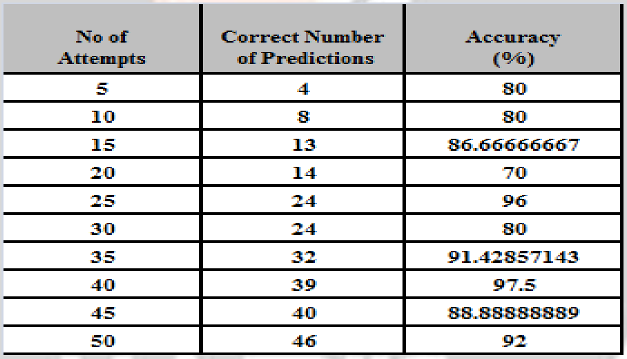
Evaluation of Accuracy – The Hybrid Recommendation system has been analyzed for its accuracy based on the user ratings and the recommendations offered. Due to the fact that the end user is the most appropriate judge for this type of system.

The evaluation has been by providing a score for the recommendation provided to the student based on the various tests given as input from the student, such as the General Knowledge test, Cognitive Ability test, Academic Test and Personality test. The user provides the score for the recommendation, where a score of more than 0 is considered as a like and all the scores below 0 are considered as a dislike. Equation 3 helps evaluate accuracy quite easily.

$$\text{(Ra)} = \frac{Currectnumberof Predictions}{TotalNumberofattempts} \times 100$$

Where, Ra= Recommendation Accuracy

The values calculated from Equation 3 are then tabulated for the estimation of the accuracy of the system in the table below.



No of Attempts	Correct Number of Predictions	Accuracy (%)
5	4	80
10	8	80
15	13	86.66666667
20	14	70
25	24	96
30	24	80
35	32	91.42857143
40	39	97.5
45	40	88.88888889
50	46	92

Table 1: Accuracy of the model

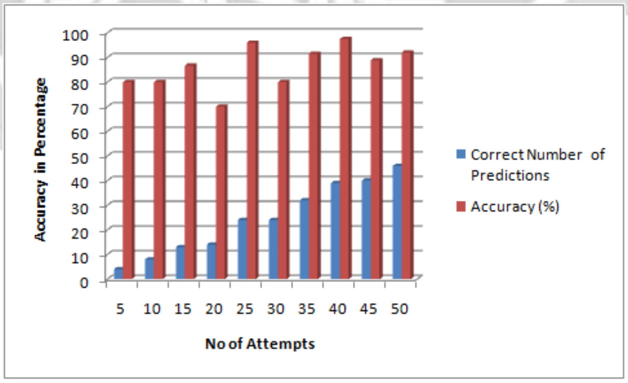


Figure 2: Accuracy evaluation of the Hybrid Recommendation System

The Graph given in the Figure above demonstrates the presented Hybrid Recommendation system's accuracy which has been calculated to be about 86.24% approximately. The figure also depicts the number of attempts and its corresponding increase in accuracy. The system is also highly consistent and steady.

The consistency of the system can be attributed to the fact that the Hybrid recommendation model has been supplemented with a semi-supervised model which accommodates the Collaborative Filtering technique with the addition of Hidden Markov Model.

V. CONCLUSION AND FUTUREWORK

The presented technique for a Hybrid Recommendation system for students in the modern educational system. The suggestions provided by the Hybrid Recommendation system have been calculated by the evaluation of the student's answers on a test. The various techniques have been elaborated in this paper in depth for the recommendation system that is supplemented with Collaborative Filtering and Hidden Markov Model with the addition of the K Means clustering to provide it with some stability and consistency. The proposed model illustrates that due to the addition of a semi-supervised learning technique to the hybrid Recommendation system increase the accuracy of the system by a large margin.

In the Future, the proposed system can be utilized on a web platform as well as an application for handheld devices and smart phones. This would provide a recommendation to the students on the go to enrich their lives and save valuable time. The system can be further developed as an API for hassle-free integration in various projects.

REFERENCES

[1] G. Kaushik Ram, N. Sai Kiran and S. Sudha, "A Mail Based Recommender System", DOI: 978-1-5090-5905-8, pp.75-81 2017 IEEE.

[2] Hualong Ma, Xiande Wang, Jianfeng Hou and Yunjun Lu, "Course Recommendation Based on Semantic Similarity Analysis", IEEE, 2017.

[3] Suleyman Uslu, Can Ozturan and Mehmet Fatih Uslu, "Course Scheduler And Recommendation System For Students", DOI: 10.1109/ICAICT.2016.7991812, IEEE.2016.

[4] Carolina Mejia, Sergio Gomez, Laura Mancera and Sibylle Taveneau, "Inclusive Learner Model for Adaptive Recommendations in Virtual Education", DOI 10.1109/ICALT.2017.101, IEEE,2017.

[5] Elham S.Khorasani, Zhao Zhenge, and John Champaign, "A Markov Chain Collaborative Filtering Model for Course Enrollment Recommendations", DOI: 978-1-4673-9005-7/16/, ,IEEE, 2016.

[6] Anirudh Kashi, Sachin Shastri, Akshay R. Deshpande Jawahar Doreswamy and Gowri Srinivasa, "A Score Recommendation System Towards Automating Assessment In Professional Courses", DOI 10.1109/T4E.2016.35,IEEE, 2016.

[7] H. Slimani, N. El faddouli, R. Benslimane, and S. Bennani Rime, "Personalized Search and Recommendation in a Digital Educational Resources Repository: The Case Of Ori-Oai", DOI: 978-1-5090-0751-6/16 ,IEEE ,2016.

[8] Muhammad Fahim Uddin, Soumita Banerjee, and Jeongkyu Lee, "Recommender System Framework for Academic Choices", DOI 10.1109/IRI.2016.70,IEEE, 2016.

[9] David Simkins and Adrienne Decker, "Examining the Intermediate Programmers Understanding of the Learning Process", 978-1-5090-1790-4/16/, IEEE, 2016 .

[10] Kathiravelu Ganeshan and Xiaosong Li, "An Intelligent Student Advising System Using Collaborative Filtering", DOI: 978-1-4799-8454-1/15/, IEEE.,2016.

****