Using Recommendation System to help Students choose a career field based on their Interests

Prof. Surendra Babu KN
REVA University, Bengaluru, India
surendrababukn@reva.edu.in

Shivendra Saurav
REVA University, Bengaluru, India
shivendrasaurav@gmail.com

Shubham Kumar Giri
REVA University, Bengaluru, India
shubhamwwe9@gmail.com

Shiwani
REVA University, Bengaluru, India
shiwani99000@gmail.com

Shivani Sharma
REVA University, Bengaluru, India
bnbindhu0@gmail.com

Abstract

Several researchers study recommendation systems to assist users in the retrieval of relevant goods and services, mostly used in e-commerce.

Several researchers study recommendation systems to assist users in the retrieval of relevant goods and services, mostly used in e-commerce.

However, there is limited information of the impact of recommender systems in other domains like education. Thus, the objective of this study is to summarize the current knowledge that is available as regards recommendation systems that have been employed within the education domain to support educational practices.

In our Experiment, we are trying to develop a questionnaire to test and know what career field the user (the person who is solving the questionnaire) is interested in. The result is dependent on variety of different factors which range from what level of study are they currently doing (PU, UG, PG, etc.) to what kind of skills they possess. All these factors are then taken in consideration and then the inputs are compiled to provide a result.

We are using Flask (a python microkernel as the backend of our questionnaire to handle all the inputs we are receiving and then Using HTML, CSS and JavaScript to display the UI/UX of the questionnaire. This will be explained in detail in later sections.

Introduction

The prototypical use case for a recommender system occurs regularly in e-commerce settings. A user, Jane, visits her favorite online bookstore. The homepage lists current bestsellers and a list containing recommended items. This list might include, for example, a new book published by one of Jane's favorite authors, a cookbook by a new author and a supernatural thriller. Whether Jane will find these suggestions useful or distracting is a function of how well they match her tastes. Is the cookbook for a style of cuisine that she likes (and is it different enough from ones she already owns)? Is the thriller too violent? A key feature of a recommender system therefore is that it provides a personalized view of the data, in this case, the bookstore's inventory. If we take away the personalization, we are left with the list of best-sellers – a list that is independent of the user. The aim of the recommender system is to lower the user's search effort by listing those items of highest utility, those that Jane might be most likely to purchase. This, of course, is beneficial to Jane as well as the e-commerce store owner. Recommender systems research encompasses scenarios like this and many other information access

environments in which a user and store owner can benefit from the presentation of personalized options. The field has seen a tremendous expansion of interest in the past decade, catalyzed in part by the Netflix Prize (Bennett & Lanning, 2007) and evidenced by the rapid growth of the annual ACM Recommender Systems conference. At this point, it is worthwhile to take stock, to consider what distinguishes recommender systems research from other related areas of research in artificial intelligence, and to examine the field's successes and new challenges. Systems that retrieve and filter the data through content and similar profiles are known as recommendation systems (RS). These systems are usually used within the e-commerce domain. For example, some websites, such as Amazon, through the application of RS allow offering the user recommendations for products that users do not know and could be of their interest. Suggested recommendations help to overcome the distressing search problem for the user. But this technology is not only used to sell products, but it is also used to suggest videos (YouTube), movies (Netflix), friends (Facebook), among others.

This demand spans across several domains, among which is the

educational domain. RS, which are applied in education, have the role of supporting teaching and learning activities through enhanced information retrieval. Nevertheless, there is limited information of the application of recommender systems in educational environments. Consequently, this study aims to summarize the current knowledge that is available concerning RS that have been employed to support educational practices.

Recommendation Systems

Recommendation Systems are software tools and techniques that provide recommendations for available products for a consumer. The recommendations apply to various aspects of decision taking, such as what things to purchase, what music to listen to or what online news to read. Recommendation Systems are generally used to provide personalized experience to users for any given product.[1]

The general term "item" is used to denote what the program recommends to users. A Recommendation Program usually focuses on a specific category of item (e.g. Books or articles) and thus its architecture, its graphical user interface and the key recommendation methodology used to produce the recommendations are all designed to provide practical and efficient suggestions for that particular type of item.

A Recommendation System is a relatively new technique used to provide these personalized experiences. Due to this, the usage of recommendation system is still very slim in practical applications, while there is still research is going on about how to make recommendation systems easier to implement and increase their efficiency.[2]

Recommendation systems are used in a number of ways, and are most widely known as playlist generators for video and music services such as Netflix, YouTube and Spotify, product recommendations for services such as Amazon, or content recommendations for social media sites such as Facebook and Twitter. Such systems can work with a single input, such as music, or multiple inputs within and across channels such as news, books, and queries for searches. [3]

Approaches

Filtering is the method which is used to filter out likes or dislikes of a user based on past experiences of the user. This is the main approach we go with while building a Recommendation System. Also, there are many types of recommendation system which will be explained briefly here in this topic.

1. **Collaborative Filtering:** It is one of the most widely known approaches to design Recommendation System. Collaborative filtering or recommender systems use a user preferences database to predict additional topics or items that a new consumer may like. In a collaborative filtering approach, a unique profile is made for every consumer, Products consumed by each consumer is helped to enhance his/her profile. Then the past experiences of a person (products consumed) from such user profiles are taken in account to recommend products to new or concurrent users.[4]

2. **Content Based Filtering:** Another common approach is content-based filtering when developing the recommender systems. Content-based filtering approaches are based on a description of the item and a user preferences profile. Such approaches are ideally suited to circumstances where information about an object (name, place, context, etc.) is known, but not about the consumer. Recommenders based on content view recommendations as a user-specific classification issue, and learn a classifier based on product features to the user's likes and dislikes.[5]

3. **Multi Criteria Recommendation System:** Multi-criteria recommended systems (MCRS) can be described as recommended systems that integrate multiple criteria preference information. Instead of designing a recommendation methodology based on a single parameter value, the overall preference of consumer u for item I these systems attempt to predict the ranking of unexplored items u by making use of preference information on various factors that influence this overall preference value.[6]

4. **Risk Aware Recommendation System:** Most of the current solutions to the recommended frameworks concentrate on providing the most appropriate material to users using qualitative information, but do not take into consideration the possibility of upsetting users with inappropriate updates. It is necessary to recognize the possibility of disturbing the customer by making suggestions under such situations, such as during a formal conference, early morning or late at night. Therefore, the efficiency of the recommendation framework depends in part on the degree to which the risk has been integrated into the recommendation process.[7]

5. **Hybrid Filtering:** Hybrid Filtering is also a method of filtering which combines all the previous filtering approaches which have been discussed above to design a Recommendation System. Hybrid filtering is normally more effective way to build a recommendation system as using this approach, we can provide recommendations based on past experiences of similar users and what consumers generally think about a product.

How to use Recommendation System in Career Improvement

Recommender system is a computer program build with the help of experts where the details of the students and their aptitudes help finding a right course for his future. This project proposes feasible predictions for student's field selection based on their marks and choice of interest. Choosing a right field in CSE/IT stream is very important for his/her future. If the decision went wrong it will be a mismatch between student aptitude, capability and personal interest. This project also reveals the research process for preparation of such a recommender system. Smart Career Guidance Recommendation System is developed for recommending skilling courses and certification courses in the CSE/IT domain. A substantial amount of literature focuses on predicting student performance in solving problems or completing courses. Many Machine learning techniques, such as decision trees artificial

neural networks, matrix factorization, collaborative filters and probabilistic graphical models, have been applied to develop student performance prediction algorithms. In this paper, we identify and apply the suitable algorithms for Student specific skill-oriented course recommendation system in the CSE/IT domain. We present the dataset built using the questionnaire and skill tests to extract the information regarding their interests, abilities This project presents Student Career Guidance and Recommendation System using the inherent student skills for choosing right career. Choosing a right career by is significant due to the diversified human abilities. Many students are choosing their career path without receiving proper advice from suitable professional or university services. This may potentially cause mismatch between academic achievements, personality, interest and abilities of the students. In order to recommend students in career selection, it is essential to build a recommendation system that provides direction and guidance to students in choosing their career. The key challenge in this project is selecting key attributes/skills that help in predicting the right path to meet diversified students' goals. System is developed using machine learning algorithms like Decision tree Classifier, Support Vector Machine, Ada Boost, Linear Discriminant Analysis, Logistic Regression…etc. Logistic Regression and Linear Discriminant Analysis has given more accurate results. We took dataset from 1000 student's data. The recommendation system will be helpful for establishing good Institution Student relationship and improving Institution reputation. This system in future we can create effective web application that can gather information by evaluating and examining. Analytical, Memory Based, Technical, Logical, Hobbies, interests in Technical/Non-Technical, Performance of the student from the childhood and skill-based tests can be conducted and information collected can be used to improve the accuracy. The Dataset can be built from several thousands of student's data. We can try to use the clustering methods for better understanding. We can also implement the techniques like Deep Neural Networks and Time series Analysis.

Implementation Using Flask

While working on this study, we developed an application, which works on Flask Microkernel, it is a

lightweight WSGI web application framework. It is designed to make it fast and easy to get going, with the potential to scale up to complicated applications. It began as a basic tool and jinja wrapper and became one of the most common Python web application frameworks.

For our application, we used Flask as backend, deployment server, while we developed the front-end User Interface and User Experience using HTML (Hyper Text Markup Language), CSS (Cascading Style Sheets) and JavaScript ES6. There are two main parts of our Project which we will focus on.

Questionnaire

The questionnaire involves of three steps. All the three steps have different types of questions which can be used to compile the results. Now the first step involves of questions which don't affect the final result. It contains of Name, Email Address and LinkedIn Profile Link of the. This information is collected to send the result to provided email address (Not implemented yet). Now after this, in the second step of questionnaire consists of questions which slightly affect the final result, which are Age, The current level of Study currently

being pursued, and how many years are still left for completion of the Study currently being pursued. Now in the third step of Questionnaire, questions which would affect the result greatly are asked. It consists of what Soft Skills does the participant possess, which field greatly interests the participant, and at last does the participant possess any specific skills. After all this data is entered, it is compiled, and a result is produced.

Compilation

Once all the data has been entered, a Recommendation System uses all the data and compares it with all the data entered by previous participants and some hard rules are considered to give a career choice which is most suitable for the participant. For example, if a person is 21 years old, and is pursuing a UG course it is most likely that the person doesn't have much experience in any field so he/she must be provided with a career choice which doesn't require any specialization. After this the soft skills help to determine what kind of role the participant can handle, and then it is decided which field is good for the participant to work in which greatly depends on past results of other participants. Now after this, the skills possessed by the

participant come into play for example if a participant possesses skills such as HTML, CSS, JS then the participant can either go for a career in Mobile and Web Development or, Front End Development. So, based on previous results Web Development would be selected and displayed as the most suitable career choice for the participant.

How to run the program

To run the program, a user should be using a Linux or Windows system and should have Python 3.x Installed. If this requirement is fulfilled, then the project can be downloaded using [this](#) link. Now the second requirement is Flask should be installed in the system, to do so, the following command should be run. [8]

```
pip install flask
```

Now after this, project folder must be opened in the terminal, or command prompt and these commands should be run to open the project in deployment server. [9]

For Windows

```
set FLASK_APP = "app.py"
```

```
flask run
```

For Linux

```
$ export FLASK_APP=hello.py
```

```
$ flask run
```

Once done, a local address will be provided. This address should be visited in a browser window to access the deployment of the project.

Conclusion

In conclusion, the outcomes of this study will benefit the students in their career selection in which the process will become easier, flexible and faster. This is because self-testing can be done without the need of comprehensive mentoring by the counsellor. Furthermore, this study also measures student's skill strengths, abilities and personality

References

1. Francesco Ricci and Lior Rokach and Bracha Shapira, Introduction to Recommender Systems Handbook, Recommender Systems Handbook, Springer, 2011, pp. 1-35

2. Goldberg, D., Nichols, D., Oki, B.M., Terry, D.: Using collaborative filtering to weave an information tapestry. Commun. ACM 35(12), 61–70 (1992)

3. Pankaj Gupta, Ashish Goel, Jimmy Lin, Aneesh Sharma, Dong Wang, and Reza Bosagh Zadeh WTF:The who-to-follow system at Twitter, Proceedings of the 22nd international conference on World Wide Web

4. John S. Breese; David Heckerman & Carl Kadie (1998). Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the Fourteenth conference on Uncertainty in artificial intelligence (UAI'98). arXiv:1301.7363.

5. D.H. Wang, Y.C. Liang, D.Xu, X.Y. Feng, R.C. Guan(2018), "A content-based recommender system for computer science publications", Knowledge-Based Systems, 157: 1-9

6. Lakiotaki, K.; Matsatsinis; Tsoukias, A (March 2011). "Multicriteria User Modeling in Recommender Systems". IEEE Intelligent Systems. 26 (2): 64–76. CiteSeerX 10.1.1.476.6726. doi:10.1109/mis.2011.33.

7. Bouneffouf, Djallel (2013), DRARS, A Dynamic Risk-Aware Recommender System (Ph.D.), Institut National des Télécommunications

8. Flask Installation - https://flask.palletsprojects.com/en/1.1.x/installation/#install-flask

9. Flask Quickstart (Get started with a Simple Flask Project) - https://flask.palletsprojects.com/en/1.1.x/quickstart/#a-minimal-application