Implementing a Recommender System for CS Undergraduate Students using Machine Learning

by

Umama Sumlin Tasnuva
14241008
Azmanul Abedin Aumi
14201036
Shariful Islam Shishir
18341011

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science

Department of Computer Science and Engineering
Brac University
April 2019

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Umama Sumlin Tasnuva
14241008

Azmanul Abedin Aumi
14201036

Shariful Islam Shishir
18341011

Approval

The thesis titled "Implementing a Recommender System for CS Undergraduate Students using Machine Learning" submitted by

1. Umama Sumlin Tasnuva (14241008)
2. Azmanul Abedin Aumi (14201036)
3. Shariful Islam Shishir (18341011)

Of Spring, 2019 has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science on April 25, 2019.

Examining Committee:

Supervisor:
(Member)

Hossain Arif
Assistant Professor
Department of Computer Science and Engineering
BRAC University

Program Coordinator:
(Member)

Jia Uddin,PhD
Associate Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Md. Abdul Mottalib
Professor and Chairperson
Department of Computer Science and Engineering
Brac University

Abstract

E-learning is a learning process that accesses educational curriculum using electronic technologies and can be operated anywhere in the world where there is facility to connect to the internet. Traditional teaching practices of face-to-face mentoring are being replaced by the non-concrete classroom where instructors and students can interact without any barriers. Prerecorded videos, eBooks (Electronic books in pdf form), short written lessons, live sessions, video calling are the main sources of E-learning. Lecturers can grade students' performance through virtual assignments and tests. Students may even opt for a degree certificate after completion of the course that are no less worthy than a degree from any renowned physical institution. In this research paper, the study of E-learning is divided in two parts. Firstly, a survey was conducted on undergraduate students enrolled in Department of Computer Science and Engineering, BRAC University. After performing statistical data mining on students' reviews, more useful information were interpreted that include the preferred online source to study, satisfaction extent on current obtainable resources, any suggestions that could make their E-learning process effortless and complaints against current online accessibility of course materials. Whether the students want an E-learning Recommender System was also deduced after this assessment. Secondly, an E-learning Recommender System containing video tutorials was built using content-based filtering, item-based collaborative filtering and user-based collaborative filtering. This recommender system was built using tools and libraries of Python programming language which contains massive resources for major CSE courses offered in BRAC University. The system has also attempted to eliminate the problems attained from the initial portion of the research. Ultimately, we proposed a hybrid filtering approach for our video recommender system considering our experimental results carried on the particular demography which revealed accuracy of the three used algorithms as 88%, 80% and 80%.

Keywords: e-learning; Recommender System; Content Based Filtering; Item-Based Collaborative Filtering; User-Based Collaborative Filtering.

Dedication

To our supporting faculty body, seniors and well wishers of the department and beyond. Love goes out to our friends and family for giving us the latent energy we always needed to get through this.

Acknowledgement

Throughout our incredible one year long journey, we have received uncountable assistance and contribution from many well-wishers. The journey would have been incomplete and vague without their constant support and priceless contribution. Firstly, we want to express our wholehearted gratitude to our thesis supervisor Hossain Arif, Assistant Professor and co-supervisor Dilruba Showkat, Lecturer- both from Department of Computer Science and Engineering of BRAC University. Their persistent motivation, guidance and expertise fueled our research progress.

Additionally, we are grateful to those students of CSE department, BRAC University who have helped us to collect the data which we used for our research work. Alongside, we are thankful to the people who are trying to make e-Learning easier, better and also fruitful by making different categories of online video tutorials.

Nonetheless, we would like to show appreciation to the Department of Computer Science and Engineering, BRAC University for providing us with all the fundamental help.

Table of Contents

List of Figures

List of Tables

Nomenclature

The next list describes several symbols & abbreviation that will be later used within the body of the document

$\sum$ The process of summing something up

θ theta is used to represent an angle

ACM Association for Computing Machinery

APA American Psychological Association

$CGPA$ Cumulative Grade Point Averages

CS Computer Science

MLE Maximum Likelihood Estimation

$PARC$ Palo Alto Research Center

$PEL - IRT$ Personalized E-Learning system using Item Response Theory

$PEOU$ Perceived Ease of Use

PU Perceived Usefulness

$SICS$ Swedish Institute of Computer Science

SIF Social Information Filtering

TAM Technology Acceptance Model

TEL Technology Enhanced Learning

$TF - IDF$ Term Frequency and Inverse Document Frequency

UI User Interface

Chapter 1

Introduction

The thought of recommender system is not a new idea but the thought of its development depending on various genres, is a new concept to us. From time to time, we face the necessity to improve this system to incorporate with the requirements of different groups of users. The software based tools and techniques which are used to provide appropriate suggestions for items or services to a user are called recommendation systems.

Now a days, recommender systems are used in almost every sector of digital platforms. For an instance, a user is searching for a particular astronomical book in Google. Nonetheless, if he finds his desired item or not, this activity of the user will be monitored by his social media accounts or other web accounts that are connected to that particular Google account. Afterwards, the user will be presented with the advertisements of similar books on his Facebook newsfeed, when visiting other websites or just while merely searching something on the internet. To sum up, this is a digital marketing strategy which uses recommendation systems' technologies. Correspondingly, in terms of music, a listener may prefer jazz while another might prefer blues. The primary thought of recommender system would have been to refer jazz music to a jazz lover from a gigantic list of all kinds of music. The recommender systems also plays a vital role in education sector.

At present, almost every enormous sites are using these recommendation systems to offer enhanced services to their users due to the high growth rate and reputation of recommendation systems. To illustrate, Amazon, YouTube, Facebook all are using recommendation systems. For the users' proper utilization of the sites, they are using these technologies to recommend their products and services in an appropriate way. At online retailers like Amazon or similar vendors, the vital uses of recommender system can be seen. Through the analysis of prior online activities of the users, Amazon presents suggestions of products or services to the users which they might get fascinated to buy. Instead of generating random suggestions, the suggestions are produced based on decisions taken by similar users. In terms of news services, they usually classify interesting news for the similar group of users.

Now the question is, how to discover these similarities. One criteria can be to find the similarity of important words in the documents. Another can be based on the readers' activities possessing the same reading perceptions. In terms of blogs, videos these same principles are being applied for recommendation. However, another way to recommend can be by evaluating user ratings. As an example, Netflix uses user ratings for offering their customers recommendations of movies that they might be willing to watch.

The main objective of recommendation system is to make decision-making process easier by suggesting the right item to the right user. It can assist people in their day-to-day actions in a number of ways like – purchasing items, watching videos, listening to music, dining at restaurants or cafés, selecting holiday destinations, reading online news etc. In our research, the general term "items" has been used to denote tutorial videos. Consequently, we can say that, recommender system is one kind of subclass for information filtering system.

1.1 Motivation

In Bangladesh, students are habituated of receiving assistance from private tutors and coaching institutes along with academic guidelines of teachers in schools and colleges. When the students move forward in life for higher studies in Universities, they find it difficult to study on their own after the usual three credit hours lecture per week for each course. Besides the class lectures and instructions from the professors, students often need extra guidance to grasp new concepts in depth. So, they seek the help of online materials, especially different e-learning websites to grab the knowledge. Unfortunately, it is often tiresome to find the suitable websites dedicated for educational purpose.

The most popular source to students for e-learning is YouTube as it is user friendly and cost free whereas most of the good websites have a membership charge. YouTube is also widely used by students because the video tutorials give them the scope of learning through visualization of the materials along with the audible guidance of the video maker. Moreover, watching a video tutorial saves more time than reading an e-book, article or presentation slides. Being a global platform, here students can find videos of their native languages along with foreign languages. Students can choose their required video according to their fondness from enormous list of tutorials.

Despite all the effectiveness, one of the main problem of YouTube is that it is not a dedicated educational web platform so it recommends many irrelevant videos as well such as movies, songs, news and many others. Thus, students get distracted and they also need to spend a bit more time to surf through all the recommended videos to find study materials. As we are undergraduate students ourselves, we felt it would have been really beneficial if there was a website which implemented recommendation systems in order to recommend only educational video tutorials. As a result, we were inspired to build an e-learning recommender to help the future students. Therefore, this was the core motive of our research.

1.2 Objective

The foremost purpose of our research is to make academic life easier for future students. Thus, through our research we have tried to find out the efficiency of recommendation algorithms and how they can be implemented to get the best outcome. Besides, we have also attempted to understand the strengths and weaknesses of every algorithm which were used in our recommendation system. In our research, we also compared results of different algorithms to find the efficiency and completeness of those. The assessment of these algorithms had led to understand which fits our requirement the most and what changes or improvements of these algorithms would be better. In addition, we have also tried to make the existing results more efficient by using our own techniques.

1.3 Thesis Outline

The remaining fragments of our thesis report have been structured in the following way:

Chapter 2 contains the literature review. It is divided into two sections. In section 2.1 there are brief discussion about the history and background of the related field. Section 2.2 has comprehensive study of past related research works on recommender systems.

Chapter 3 has information about out datasets that we used for our research purpose. This chapter has five sections. Section 3.1 says the sources of our data collection. In section 3.2 we have shown how we conducted a survey on students through eight graphs of statistical data mining. The sections 3.3, 3.4 and 3.5 have snapshots of our datasets for content based filtering method, user-based collaborative filtering method and item-based collaborative filtering method. Additionally, all features of our datasets are described briefly.

Chapter 4 is all about the methodology. It has four sections with several subsections. Section 4.1 contains the name of the algorithms which we analyzed through our datasets. The subsections 4.1.1, 4.1.2 and 4.1.3 describe the process and concepts of content based filtering method, user-based collaborative filtering method and item-based collaborative filtering method respectively. Section 4.2 is one of the most important part for our research which contains the overall design of our system through a figure captioned as 'Workflow of the system to recommend videos to user n upon login'

Chapter 5 contains three sections. In section 5.1, 5.2 and 5.3 we have shown the workflow diagram for all three filtering methods.

Chapter 6 is dedicated for result analysis. It has three sections with few subsections. For result analysis part we used confusion matrix, accuracy, precision, recall, specificity and F1 score which we described theoretically within different subsections under section 6.1. Afterwards, we have shown confusion matrices for three filtering methods within three subsections under section 6.2. The subsection 6.2.4 contains a table for overall performance measurement for used algorithms. The third section of this chapter is 6.3 which has the information regarding the comparison between the used algorithms.

Chapter 7 is the last chapter of our thesis report, divided into two sections. The first section 7.1 has elaborated description about our future work plan. Here, we have attached pictures of tentative UI of our e-learning recommender web application which can be built later using our research. The pictures show home page and video displaying page of our dedicated web application for a particular user. Lastly, 7.2 contains the conclusion.

At the very end of the research paper, all the references are provided for every citations in APA format.

Chapter 2

Literature Review

The prerequisite of conducting a successful research is to study thoroughly about the evolution history and past relevant works done on the particular sector. Likewise, we too devoted ourselves completely to instil the correct essence of recommendation systems deeply in our brains.

2.1 History & Background Analysis

Personalization was not a new thing long before the term 'Recommendation Systems' was invented. In the past, before recommendation systems had emerged, people performed personalization manually. Like, a buyer can engage a salesman to manually choose and suggest suitable dresses according to his choice or requirements of the clothing. In order to implement personalization into broader aspects, technological personalization was introduced in the form of the recommendation systems.

The foundation of recommendation systems is based on data mining, information retrieval, statistics, etc. Recommender systems are useful alternative to search algorithms since these help users to discover items they might not have found otherwise. Leskovec et al in the book [14] wrote, in 1990, at Columbia University, Jussi Karlgren introduced the term "Recommender System" for the first time through one of his technical report, named "Digital Bookshelf". Later on, he implemented "Recommender System" at scale and also continued his works on that through different technical reports from 1994 onwards at SICS. At the same time, he cooperatively worked with research groups led by Pattie Meas at MIT, Will Hill at Bellcore, and Paul Resnick also at MIT. Their amazing research work on recommender system with GroupLens received the 2010 ACM Software System Award.

The group led by David Goldberg built the first recommender system named 'Tapestry' at Palo Alto Research Center (formerly Xerox PARC). Tapestry was a revolutionary mail and repository system which was designed to recommend documents from newsgroups. It allowed users to search based on the document contents and reactions recorded from other users. Like, user could ask it to "give me all the docs containing the words 'racing bike' that the user 'William' has considered 'excellent'". Another early recommender system was RINGO. RINGO was a music recommendation system. The user could send an email with 10 albums and he liked each one to the recommender. Later, the user would receive a list of things he might like along with a prediction of how much the system thought he would like each one. Both of these recommendation systems used Social Information Filtering (SIF).

Rodríguez [34] said, the two kinds of information using which recommender systems function are

1. **Characteristic information:** This is information about features of items (keywords, categories, etc.) and users (preferences, profiles, etc.).
2. **User-item interactions:** This is information in forms of ratings, number of purchases, likes etc.

Using these, the classification of recommender systems are: the algorithm which uses characteristic information is called content-based filtering, and another algorithm which is based on user-item interactions is named collaborative filtering. The combination of both types of filtering is called Hybrid system. The targets of hybrid systems are to eliminate and reduce problems or lacking that are generated when working with just one of these.

2.2 Related Works

Previously, numerous research papers on recommendation systems has been published. In the research by Khribi et al in [6], users' profiles were analyzed based on their recent navigation history the researchers analyzed similarities and dissimilarities among user likings and the learning resources. To do this, they started by mining learner profiles using Web usage mining techniques and content based profiles using information retrieval techniques. Finally, they applied various recommendation strategies to provide relevant contents to the active learner. Furthermore, Ghauth and Abdullah [9] proposed a framework along with its prototype design. The recommender model suggested relevant resources based on similarity of content items using vector space model and good learners' ratings strategy. The average ratings of good learners were taken into account to suggest items to improve other learners' learning process.

In the research paper of Bobadilla et al [8], which given user is similar to another set of users were determined based on the similarities of preferences. To find these similarities three approaches were used: memory-based methods, model based methods and hybrid approaches. Finally, the system provided results to that user based on the group of users who had greater knowledge, that is, their ratings were more prioritized. Another research paper [26] presented by Hasan and Schwartz, a prototype recommender system was built named as RecAdvisor. The idea of this model was to assist in finding and recommending Ph.D. advisors to the students based on various criteria – interested area of research, publication records etc.

Tan et al in [7] narrated, the whole study was focused on the user-based collaborative filtering method. Data collection, data ETL (Extract, Transform and Load), model generation, strategy configuration, and service supply. In addition, they proposed an architecture with seven modules in total with four of them as core modules – recommendation models database, recommendation system database, recommendation management, data/model management. Additionally, Gunawan et al [17] wrote how a learning recommendation system affects on-screen learning and up to what extent was studied. It was basically a statistical approach where they targeted undergraduate students of an Indonesian University and ranked them according to their Cumulative Grade Point Averages (CGPAs). The students were divided into two groups: one group were learning with the supervision of recommendation system and another group without it. They performed a t-test on the result and found out that average score of learning with recommendation system was way too high than of those without it.

The research paper [23] by Tarus et al was a review of ontology-based recommender systems for e-learning. The study was divided into four parts – analyzed and classified journal papers on this topic which were published from 2005-2014, categorized different recommendation techniques formerly used in ontology-based recommenders, then categorized the knowledge representation techniques, ontology types and ontology representation languages, and discussed the future trends of this recommendation approach in the field of e-learning. The paper concluded that, both the use of ontology for knowledge representation in e-learning and hybridization of knowledge-based recommendation with other recommendation techniques can considerably boost the effectiveness of e-learning recommenders. Zhuhadar et al in [10], introduced a multi model ontology based recommendation system. This hybrid recommender system used content-based method (domain ontology model) and rule-based method (learner's interest-based and cluster-based). This proposed approach has been implemented on the HyperManyMedia1 platform.

Most of the e-recommender does not take into consideration about the learner's learning potential. Chen et al [3] have addressed this issue and proposed a recommendation model based on item response theory (PEL-IRT) which considers both the course materials' difficulty and the learner's ability to provide individual learning paths for learners. To obtain more precise estimation of learner's ability, the maximum likelihood estimation (MLE) was applied to estimate learner's ability based on explicit learner's feedback. Moreover, to determine an appropriate level of difficulty parameter for the course materials, this study also proposed a collaborative voting approach for adjusting course materials' difficulty. The research work [4] by Tang and McCalla proposed a developing e-learning system which can adapt itself both to the learners and to the open Web. They also figured out the differences of making recommendations in e-learning and other domains. Learner's interest and background knowledge were the two main aspects that they targeted. They differentiated the effectiveness between content-based and hybrid approach. The results concluded that hybrid approach could lower the computational costs without negotiating the overall performance of the recommendation system.

Sikka et al [12] suggested to use mining techniques to build an agent that could recommend online learning activities in a course web site based on learners' access history to improve navigation of course materials. In the research [19], Maravanyika et al proposed a framework which is based on an adaptive recommender system to benefit both personalized teaching and learning on e-learning platforms. Verbert et al in [13], they have analyzed currently how much work has been done on TEL recommender system and what might be the future challenges of it. The research by Klašnja-Milićević et al [28], proposed a model which uses clustering technique based on learning style model to reduce tag space to improve execution time and decrease memory usage without deteriorating the quality of recommendations. In [31], Alharbi et al conducted surveys on students in Saudi Arabia to explain and predict the usage intention of e-learning recommender system. They developed questionnaire based on an extended technology acceptance model (TAM) and their findings revealed that perceived usefulness (PU) and perceived ease of use (PEOU) are significant determinants of e-learning recommenders' system initial acceptance. Soonthornphisaj et al [5] developed a smart e-learning recommender system which they implemented at the faculty of Resource and Environment, Kasetsart University at Sri-racha campus and found that their system were highly appreciated by the instructors and learners.

In the research by Zaıane [1] has developed a recommender agent for e-learning system which has similar goals as the agent built by Sikka et al [12]. Albatayneh et al [25] studied the utilization of learners' negative ratings in semantic content-based recommender system for e-learning forum. Verma [33] applied predictive analytics to build a recommender system which recommends elective course for the student while considering student's preferences for courses. In [2], Lu proposed a framework for a personalized recommender system that included to two technologies - one is a multi-attribute evaluation method to justify a student's need and another is a fuzzy matching method to find proper learning resources according to each students' need. Ghauth et al [11] proposed a new e-learning recommender system framework that used content-based filtering and good learners' ratings to recommend learning materials. Students after using the e-learning recommender system had performed better than before. In the research by Sheshasaayee et al [32], the main focus was to understand the research done and current trends in e-learning environment.

Chapter 3

Datasets

3.1 Data Collection

To collect data for the first part of the research, we have conducted a survey among the undergraduate students of Computer Science and Engineering department in BRAC University, Bangladesh. The survey was conducted through interviews and Google forms. A total of 65 students' responses were taken. We stopped collecting more data from students because we were getting similar responses after a certain time.

For the second part of the research, few students were asked to provide their demographic information and to rate video tutorials of CS courses on YouTube. Their ratings were collected after they have viewed the entire video. The video parameters from YouTube were manually extracted instead of using a web-crawler, because to extract our few necessary parameters from the huge crawled information would have been more time consuming and tiresome. These datasets were then used to implement content based filtering method, user-based collaborative filtering method and item-based collaborative filtering method respectively.

3.2 Survey conducted on Students

The data set collected from the survey went through the process of cleaning it from any anomalous or inconsistent responses. It was transformed into more useful form by sorting out in proper columns in an excel file, which was used for statistical data representation. This statistical assessment is represented using various data representation models including pie chart, doughnut chart, histogram and bar chart.

Figure 3.1: *Flowchart of Assessment of E-Learning*

Data summarization using statistical data mining

Demographic Information

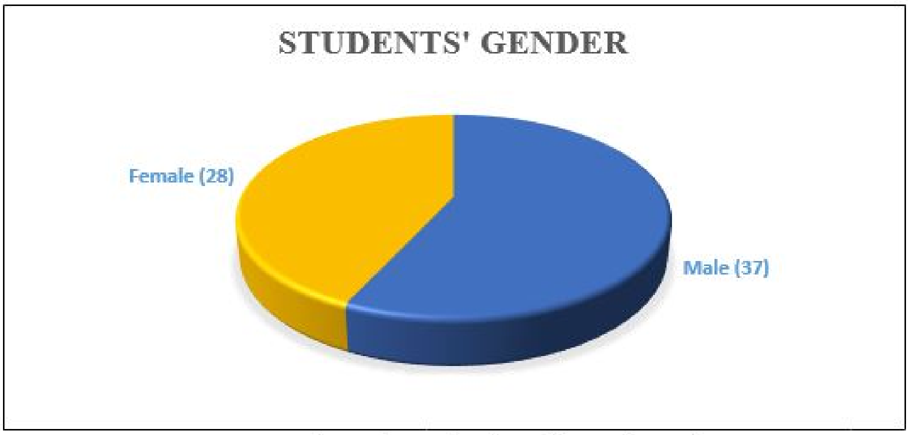

Figure 3.2: *Pie chart of No. of Male and Female Students*

Comment: 57% of the students are Male and 43% percent is Female

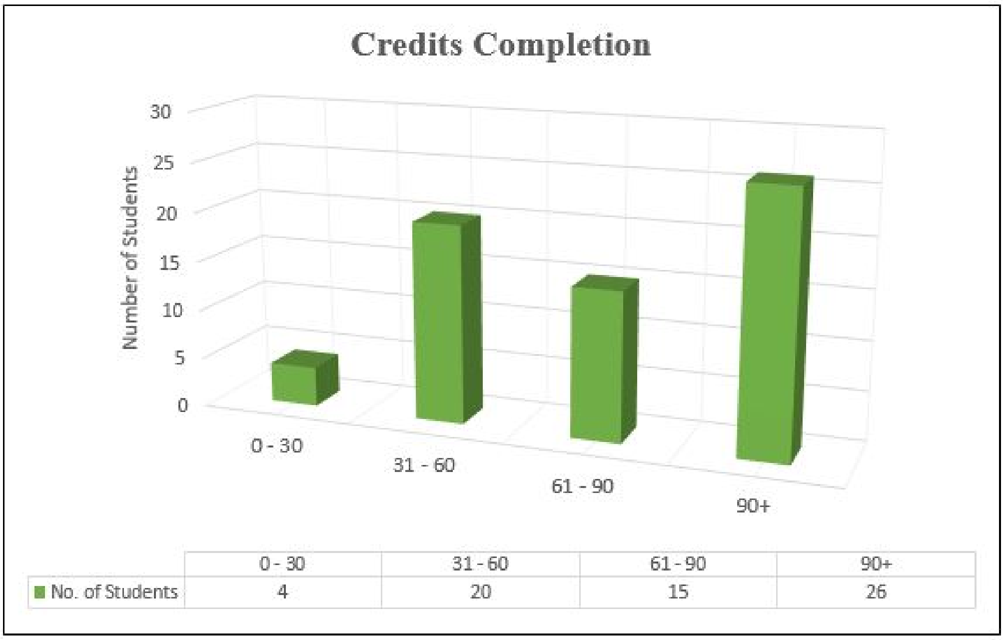

Figure 3.3: *Histogram of No. of credits completed by Students*

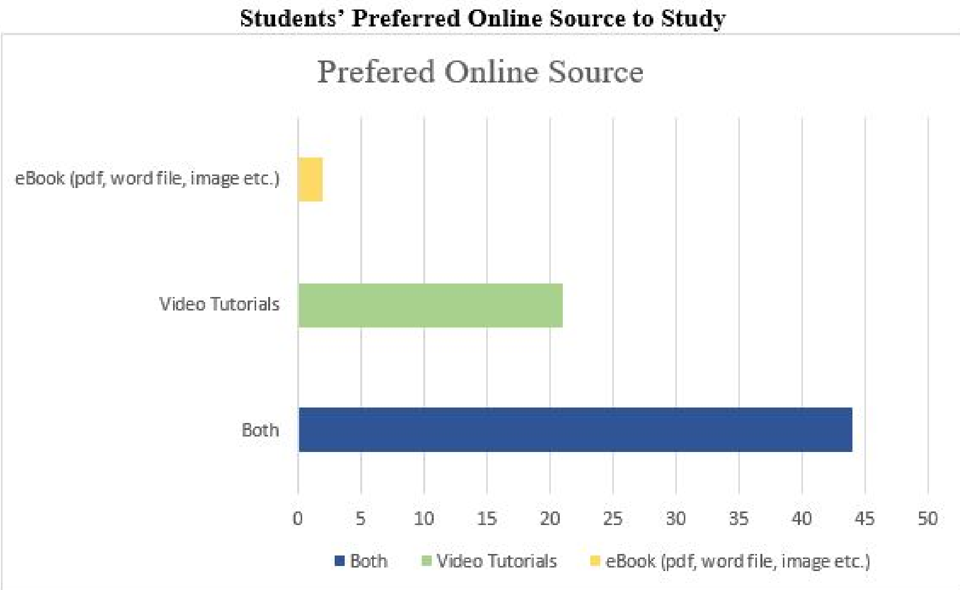

Figure 3.4: *Bar chart of Students' preferred online source to study*

Students' Satisfaction extent on TSR (University's offline server for study materials)

"Not much availability of resources"

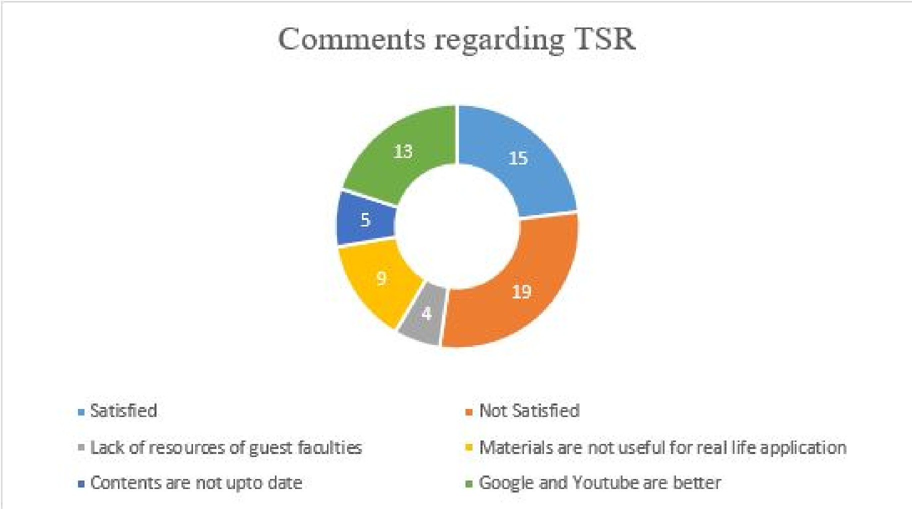

Figure 3.5: *Doughnut chart of Students' Satisfaction Extent on TSR*

Students' Complaints against Current Accessibility of Course Materials

"Difficult to find decent tutorials on some topics"

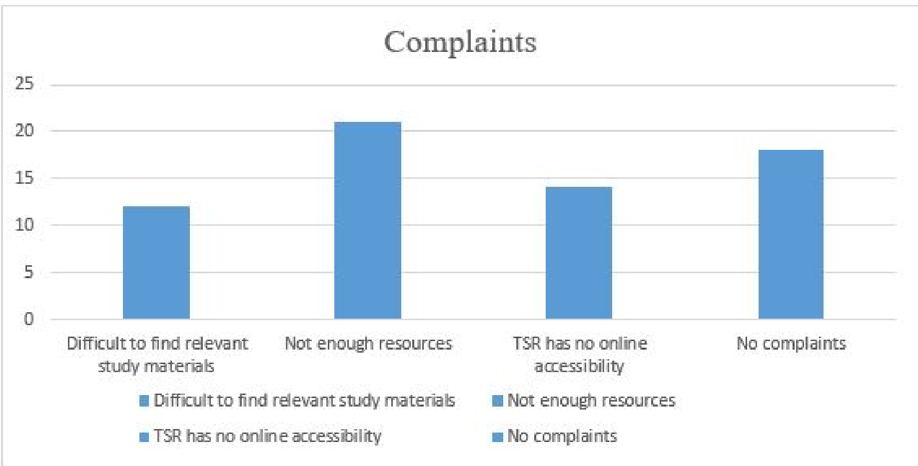

Figure 3.6: *Histogram of Students' Complaints*

Students' Suggestions to Ease E-learning

"An app that has all the resources regarding CSE courses"

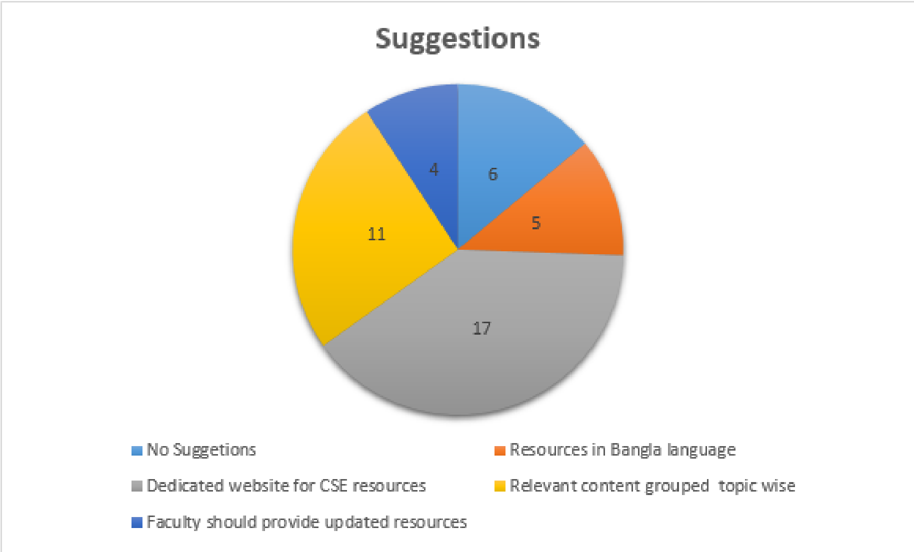

Figure 3.7: *Pie chart of Students' Suggestions*

Students' Votes for Dedicated E-learning Recommendation System

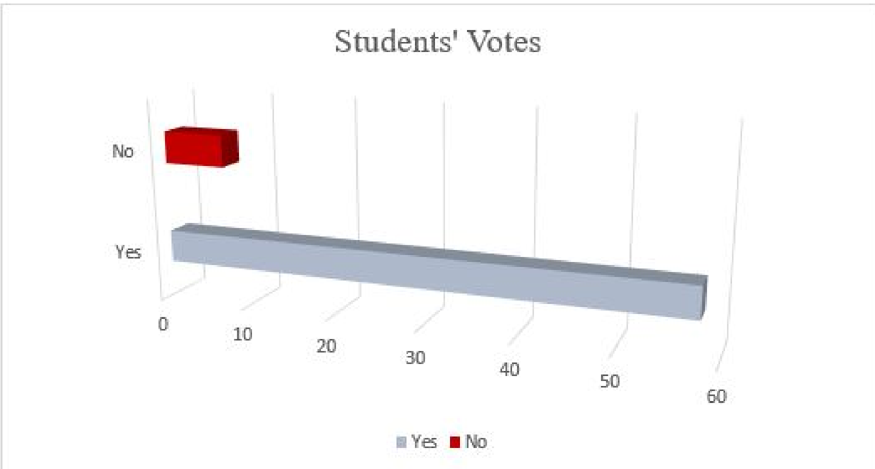

Figure 3.8: *Bar chart of Students' Votes*

3.3 Dataset for Content Based Filtering Method

The following dataset (first 5 columns are attached) is used to implement content based filtering method on user n. After implementation of the method, user n is recommended relevant videos. We have implemented this algorithm on a dataset of 200 data.

1	Course code	Course title	Topic	Video title	Duration	Views	Likes	Dislikes	Comments	Rating	Language
2	CSE221	Algorithms	Breadth First Search Algorithm	Breadth First Search Algorithm	4.33	908068	4100	331	231	9.5	1
3	CSE221	Algorithms	Breadth First Search Algorithm	Graphs: BFS - Breadth First Search Traversal	10.47	43244	0	0	18	9.5	1
4	CSE221	Algorithms	Breadth First Search Algorithm	BFS algorithm simulation in bangla	8.45	15078	114	18	21	7	2
5	CSE221	Algorithms	Breadth First Search Algorithm	AI Bangla Tutorial 1 : Breadth First Search | BFS	5.44	21879	154	13	32	8	2
6	CSE221	Algorithms	Breadth First Search Algorithm	Graph Traversals - Breadth First and Depth First	10.08	484284	5000	112	358	9	1

Figure 3.9: *Partial dataset for Content Based Filtering Method*

All the features are taken from video tutorials on YouTube:

- Course code – the video belongs to which course code
- Course title – the video belongs to which course code (corresponding to course code)
- Topic – the video belongs to which topic in the course
- Video title – video title given on YouTube
- Duration – duration of the total video
- Views – number of views on YouTube
- Likes – number of likes on YouTube
- Dislikes – number of dislikes on YouTube
- Comments – number of comments on YouTube
- Rating – rating provided by the user
- Language – language used in the video (English = 1, Bangla = 2, Hindi = 3)

3.4 Dataset for User-Based Collaborative Filtering Method

The following dataset (first 5 columns are attached) was used to implement content based filtering method on user n. After implementation of the method, user n is recommended relevant videos. We have implemented this algorithm on a dataset of 150 data.

1	Student ID	Semester	Credits	CGPA	Language	Duration
2	15301066	9	72	3.26	1	10
3	14201022	10	99	3.26	1	20
4	14101235	13	111	2.89	1	20
5	15304022	9	69	2.41	1	10
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Figure 3.10: *Partial dataset for User-Based Collaborative Filtering Method*

All the features belong to users:

- Student ID – Student ID of the student in BRAC University
- Semester – number of semester completed
- Credits - number of semester completed
- CGPA – Cumulative Grade Point Average of student
- Language – language preferred by the user for video tutorial (English = 1, Bangla = 2, Hindi = 3)
- Duration - duration preferred by the user for video tutorial [Small (0–7 min) = 10, Medium (8–15 min) = 20, Large (19+ min) = 30]

3.5 Dataset for Item-Based Collaborative Filtering Method

The following dataset (first 5 columns are attached) is used to implement user-based filtering method on user n. After implementation of the method, user n is recommended relevant videos. We have implemented this algorithm on a dataset of 200 data.

1	Course code	Course title	Topic	Video title	Rating_1	Rating_2	Rating_3	Rating_4	Rating_5	Ra
2	CSE111	Programming Language - II	Polymorphism	4.7: Introduction to Polymorphism - The Nature of Code	8		7.5		9	
3	CSE111	Programming Language - II	Polymorphism	Java Bangla Tutorials 126 : Polymorphism (Theory)		7	8			
4	CSE111	Programming Language - II	Polymorphism	Polymorphism In Java (Part 1) || Method Overloading in Java	9	7.5		8	9	
5	CSE111	Programming Language - II	Polymorphism	Core Java Tutorial | What is Polymorphism in Java ?			6		7	
6	CSE111	Programming Language - II	Polymorphism	Java Programming Tutorial - 55 - Intoduction to Polymorphism		8	6.5	7.5		
7	CSE111	Programming Language - II	Polymorphism	OOPs Polymorphism tutorial (Lecture) in java with example	6	7		7	8	

Figure 3.11: *Partial dataset for Item-Based Collaborative Filtering Method*

All the features are taken from video tutorials on YouTube:

- Course code – the video belongs to which course code
- Course title – the video belongs to which course code (corresponding to course code)
- Topic – the video belongs to which topic in the course
- Video title – video title given on YouTube
- Rating_1 – rating of user 1 for the video
- Rating_2 – rating of user 2 for the video
- Rating_3 – rating of user 3 for the video
- Rating_4 – rating of user 4 for the video
- Rating_5 – rating of user 5 for the video
- And so on till Rating_n (upto n^{th} user)

Chapter 4

Methodology

4.1 Algorithms

According to Luo [30], a recommender system makes prediction based on users' historical behaviors like many machine learning techniques. The system specially predicts user preference for a set of items based on past experiences. Classification of machine learning algorithms in recommender systems are done mainly into two categories – content based filtering and collaborative filtering method. Moreover, collaborative filtering is of two types – user-based collaborative filtering and item-based collaborative filtering. Modern recommender systems combine both of these approaches to build hybrid systems.

4.1.1 Content Based Filtering Method

Das [15] narrated, content-based filtering method is a method which recommends an item to a user that have similar features and characteristics with items that were highly rated by the user in past. Each item has an item profile, where its properties are listed in a table structure. In our case, the properties of item (video) are course code, course topic, video title, language, duration, number of views, likes, dislikes. Features are compared and item scores are collected. The most similar items are be recommended to the user based on best matching scores. This method relies on the features of the items only not on the user preference. Cosine similarity measure is mostly used to calculate the similarity between items seen by the user and items not yet seen by the user.

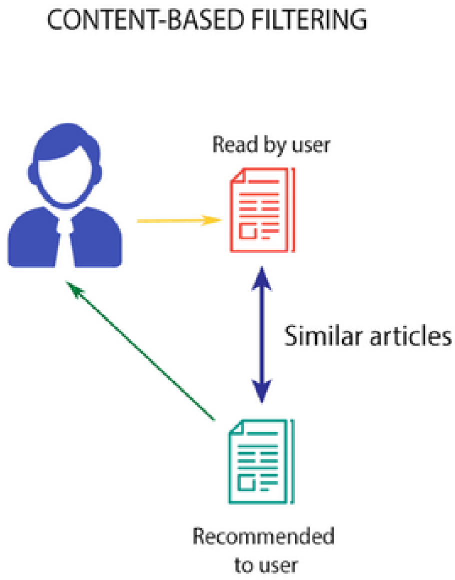

Figure 4.1: *Process of Content Based Filtering Method [16]*

Concepts used in content-based filtering method are:

1. **<u>Term Frequency (TF) and Inverse Document Frequency (IDF) (TF-IDF)</u>**

   Tf-idf stands for term frequency-inverse document frequency, and the tf-idf weight is used in information retrieval and text mining. In the book by Silge et al [21], this measurement is used to understand how often a word appears in a document and how important it is in the document. It is calculated using the given formula (4.1)
   TF, Term Frequency, is a measure of how frequently or how many times a term is present in a document.

   (a) TF(t) = (No. of times term t appears in a document) / (Total no. of terms in document)
   IDF, Inverse Document Frequency, measures how important a term is in the document.

   (b) IDF(t) = `log_e` (Total number of documents / Number of documents with term t in it)

   (c) Word that has higher tf-idf weight is more important in the document

$$w_{i,j} = tf_{i,j} \times \log(\frac{N}{df_i}) \tag{4.1}$$

$$\mathrm{tf_{i,j}} = \text{number of occurrences of } i \text{ in } j$$
$$\mathrm{df_i} = \text{number of documents containing } i$$
$$N = \text{total number of documents}$$

Formula to calculate, w, tf-idf weight of a word in a document.

2. **Cosine Similarity** - Cosine similarity is a measure of similarity between two non-zero vectors an inner product space that measures the cosine of the angle between them. To compute how much user's preference aligns with the item's vector, we will use the concept of cosine similarity. It is calculated using the given formula (4.2):

$$similarity(A, B) = \frac{A \cdot B}{||A|| \times ||B||} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \times \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{4.2}$$

Formula to calculate cosine similarity (cos θ) between vector A and B

3. **Adjusted Cosine Similarity** - Adjusted cosine similarity is another way of calculating similarity between items. A modified form of vector-based similarity is adjusted cosine similarity which is often more useful than cosine similarity for certain circumstances. The fact that different users have different preferences which results in different ratings schemes are considered in this similarity calculation method. There might be some users who rate items highly in general, and others might give items lower ratings as a preference. In order to eliminate this drawback from vector-based similarity or cosine similarity, the average ratings for each user is subtracted from each user's rating for the pair of items. It is calculated using the given formula (4.3):

$$sim(a,b) = \frac{\sum_{u \in U} (r_{u,a} - \overline{r_u})(r_{u,b} - \overline{r_u})}{\sqrt{\sum_{u \in U} (r_{u,a} - \overline{r_u})^2} \sqrt{\sum_{u \in U} (r_{u,b} - \overline{r_u})^2}} \tag{4.3}$$

- U : set of users who have rated both items a and b
- r_u : the average rating of user u
- $r_{u,a}$: the average rating of user u for item b

Formula to calculate adjusted cosine similarity between vectors A and B

After calculation of similarity, the angle between the two vectors is calculated by taking *cos* inverse of the result.

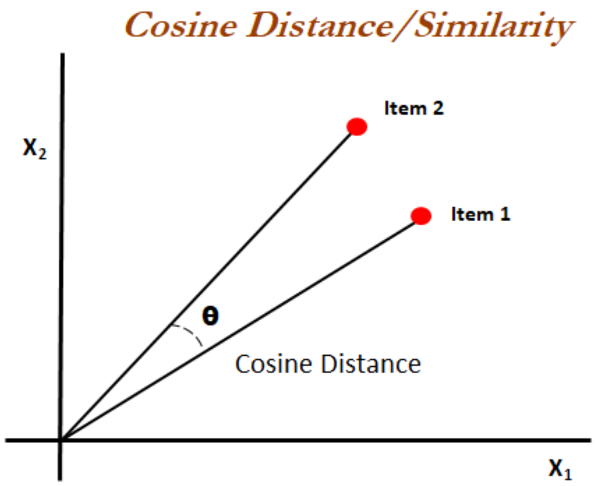

Figure 4.2: *Relation showing the cosine distance/similarity between Item 1 and Item 2 [18]*

The smaller the angle between Item 1 and Item 2, the more similar they are. More similar items has more common features or characteristics between them.

4.1.2 User-Based Collaborative Filtering Method

This collaborative filtering method uses user-user connection. To recommend items to a user, a group of similar users are deduced based on their preferences. Pinela [20] mentioned users with similar ratings or preferences are grouped together, then another user who has the same preferences as that group is suggested with the items that the group prefers. Huang [27] described that there are two types of user opinions, explicit opinion and implicit opinion. Explicit opinion directly shows how a user rates an item (rating an app or movie). Implicit opinion provides the heuristics about how a user likes an item or what features a user prefers in an item. Measuring similarity between users also consider the characteristics or demographic information of users along with their explicit and implicit opinion. Similarity between users can be calculated using Jaccard similarity or Cosine similarity. Cosine similarity is more accurate measurement so usually this is used.

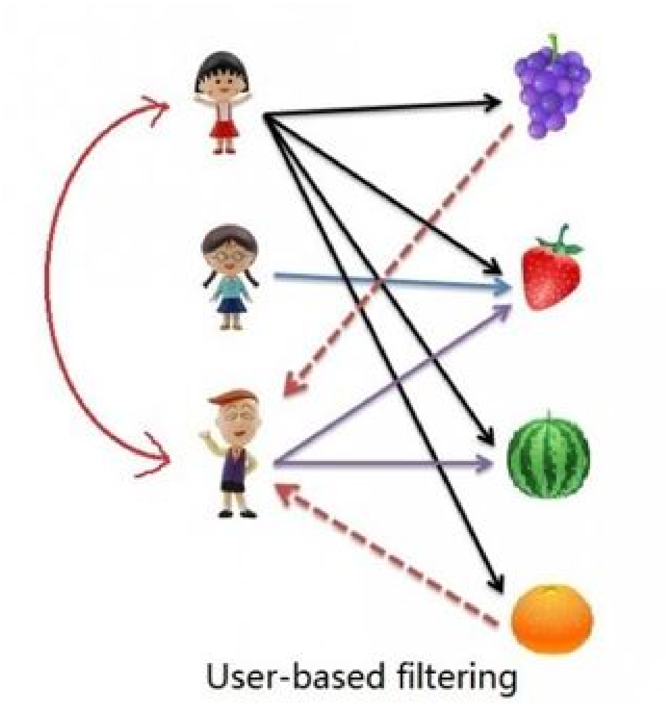

Figure 4.3: *Process of User-Based Collaborative Filtering Method [20]*

4.1.3 Item-Based Collaborative Filtering Method

This collaborative filtering method uses item-item connection. For a particular item, other items with similar features are found out. The rating for the item is estimated by calculating its similarity with the similar bunch of items. Tomar in [24] stated, for calculation of similarity between two items, the set of items the target user has rated is considered and how similar they are to the target item i is computed and then k most similar items are selected. To find the similarity of items, all users' ratings are taken in addition to the target user's ratings. Also it was written by Lew et al. [29], similarity between two items is calculated by taking the ratings of the users who have rated both the items. Then, adjusted cosine similarity measure is used for calculation. The items that have higher similarity score to target item i are grouped together as k most similar items.
Now, the prediction rating of the target item i is calculated for the user that is if the user had seen or rated that particular item i, then how much the user would have rated it. The prediction of user-item pair is calculated by taking summation of similarities between item pairs multiplied with the ratings given by user u. It is then divided with the summation of similarity scores of all similar items in k neighborhood. It is calculated using the given formula (4.4):

$$P_{u,i} = \frac{\sum_{all\ \ similar\ \ items,\ \ N}(S_{i,N} * R_{u,N})}{\sum_{all\ \ similar\ \ items,\ \ N}(|S_{i,N}|)} \tag{4.4}$$

Formula of weighted sum to predict the rating of item i

Finally, the items with higher predicted ratings are recommended to the user as those have the higher chance of being liked by the user.

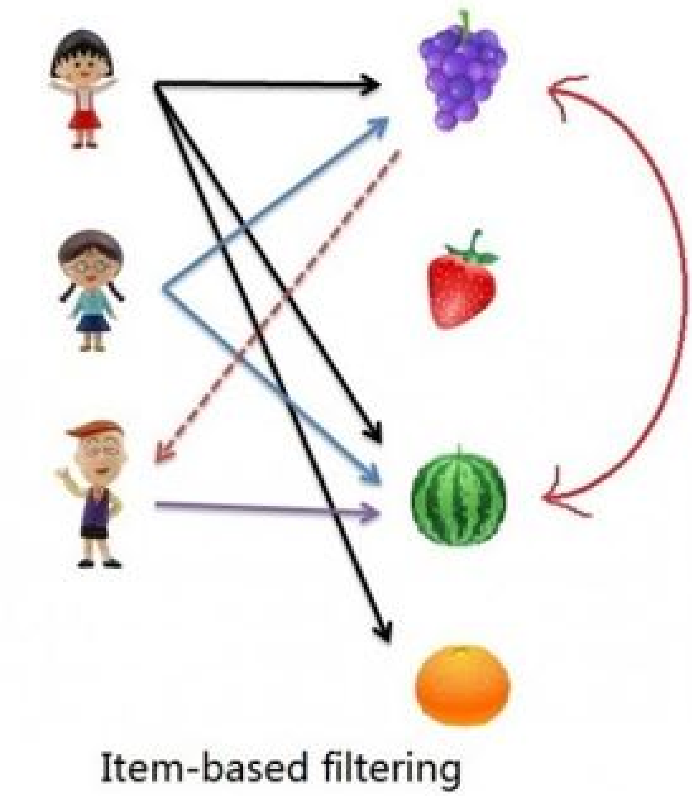

Figure 4.4: *Process of Item-Based Collaborative Filtering Method [20]*

4.2 System Design

When a user n logs in with his username and password, the system identifies the user. Then, the system applies the three machine learning techniques in recommendation system to recommend relevant video tutorials on the homepage of the user.

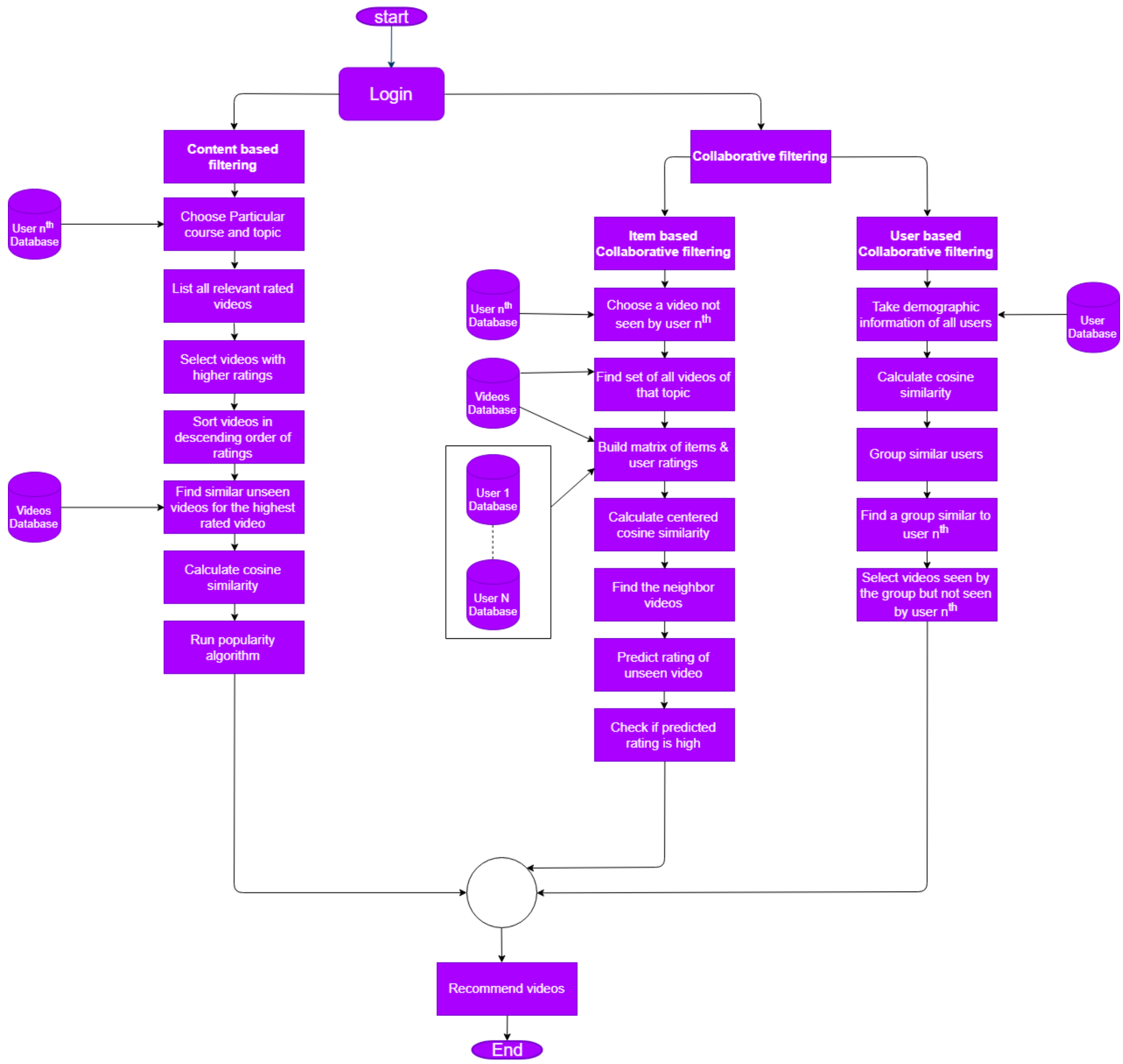

Figure 4.5: *Workflow of the system to recommend videos to user n upon login*

Chapter 5

Implementation

This chapter elucidates in details, the complete development of how all the three machine learning algorithms mentioned in previous chapter are implemented.

5.1 Content Based Filtering Method

This technique is performed on individual user's database (User n^{th} Database) and the database of all videos in the system (Videos Database). 'User n^{th} Database' contains information of the all the videos that were watched and rated by this particular user. 'Videos Database' contains all the videos seen and rated by all existing users.

Firstly, a topic and its corresponding course of a video watched by the user are chosen from 'User n^{th} Database'. All the relevant videos of that same topic and course are listed. The videos which has higher ratings (rating greater than or equal to 7 out of a scale of 10) are selected. The selection of videos are then sorted in descending order of ratings given by the user.

Secondly, from the 'Videos Database', videos that are similar (same topic and course) to the selected videos with high ratings and which are also not seen by the 'User n' are found. Cosine similarity is calculated between the unseen and seen videos. This is used to measure how similar the features of the videos are like - duration, language etc. The video pair which has higher cosine similarity score or smaller angle between them are more similar to each other. Then, 'Popularity Algorithm' is operated on those similar videos. This algorithm calculates how much viewers liked or disliked the video. If the liking percentage over the number of views (number of likes divided by number of videos) of the video is higher, then it is a more popular video. On contrary, if the disliking percentage over the number of views (number of dislikes divided by number of videos) of the video is higher, then it is a less popular video.

Finally, the unseen videos that are more popular are recommended to the user before recommending the less popular videos.

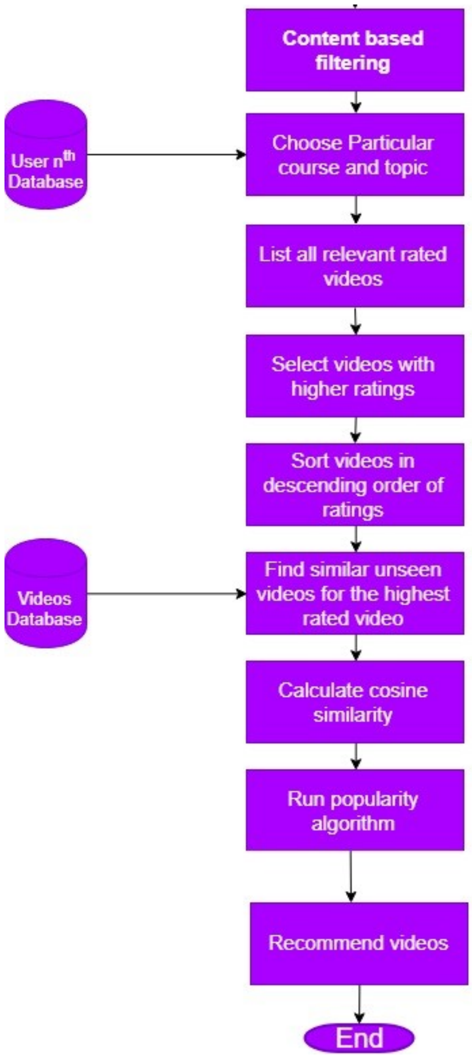

Figure 5.1: *Workflow of content based filtering method*

5.2 User-Based Collaborative Filtering Method

User-based collaborative filtering method is performed on the database of all users who are using the recommender system. 'User Database' contains all the demographic information of all the users which were taken from them upon registering or signing up for the system.

All demographic information of all users are taken and cosine similarity between users are calculated. This measurement will deduce how similar are the user's feature with those of other user's. Like, users who have completed similar number of credits, has same preference for language of the video, prefer similar length of videos etc. – are similar to each other. The user pair which has higher cosine similarity score or smaller angle between them are more similar to each other.

After calculating similarities, users are grouped with nearest k neighbors. User n is grouped with its nearest k neighbors such that all the users in that group has similar characteristics, preferences and necessities. For example – the user who has completed similar number of credits as his k neighbors, might be doing the same course as his neighbors and would need video tutorials for that particular course.

Lastly, it is checked which videos the k-nearest neighbors have watched. These information is taken from each neighbor's respected videos database. Among these videos, the videos which User n has not yet viewed yet are recommended to the user. If the user has already seen it, then it is not again recommended to him.

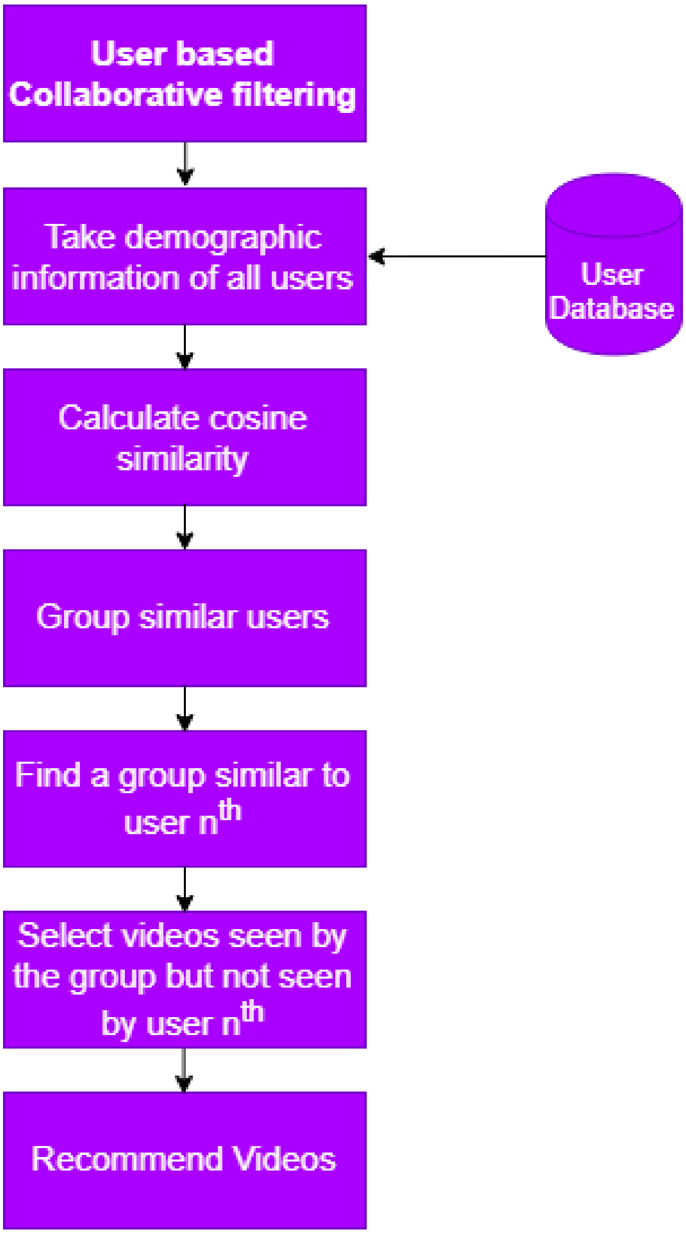

Figure 5.2: *Workflow of user-based collaborative filtering method*

5.3 Item-Based Collaborative Filtering Method

This algorithm is performed mainly on individual user (User n^{th} Database) and the database of all videos in the system (Videos Database). 'User n^{th} Database' contains information of the all the videos that were watched and rated by this particular user. 'Videos Database' contains all the videos seen and rated by all existing users. Other user's databases are also used to take their ratings.

At first, a video which has not been watched by user n is selected from 'Videos Database'. All videos in the 'Videos Database' which are of same course and topic as the selected video and has not been viewed by the user n is chosen as a list.

Then, a matrix is built which consists of items (videos) and users' ratings. The videos are the row of the matrix and users' ratings for those videos are the columns of the matrix. After building the item-rating matrix, adjusted cosine similarity between the selected video (not seen by user n) and all other videos are calculated using the ratings of all users. The videos which has higher similarity score (smaller angle) with the selected video and also which are not yet seen by the user n are taken as k-nearest neighbors of the selected video.

At last, the rating for the unseen video is predicted using the formula no. (4.4). This is a prediction of how the video would have been rated by the user if he had watched it. In this way, ratings are predicted for all unseen videos of that topic of the particular user. The video tutorials with higher predicted ratings are recommended to user before recommending the videos with lower predicted ratings.

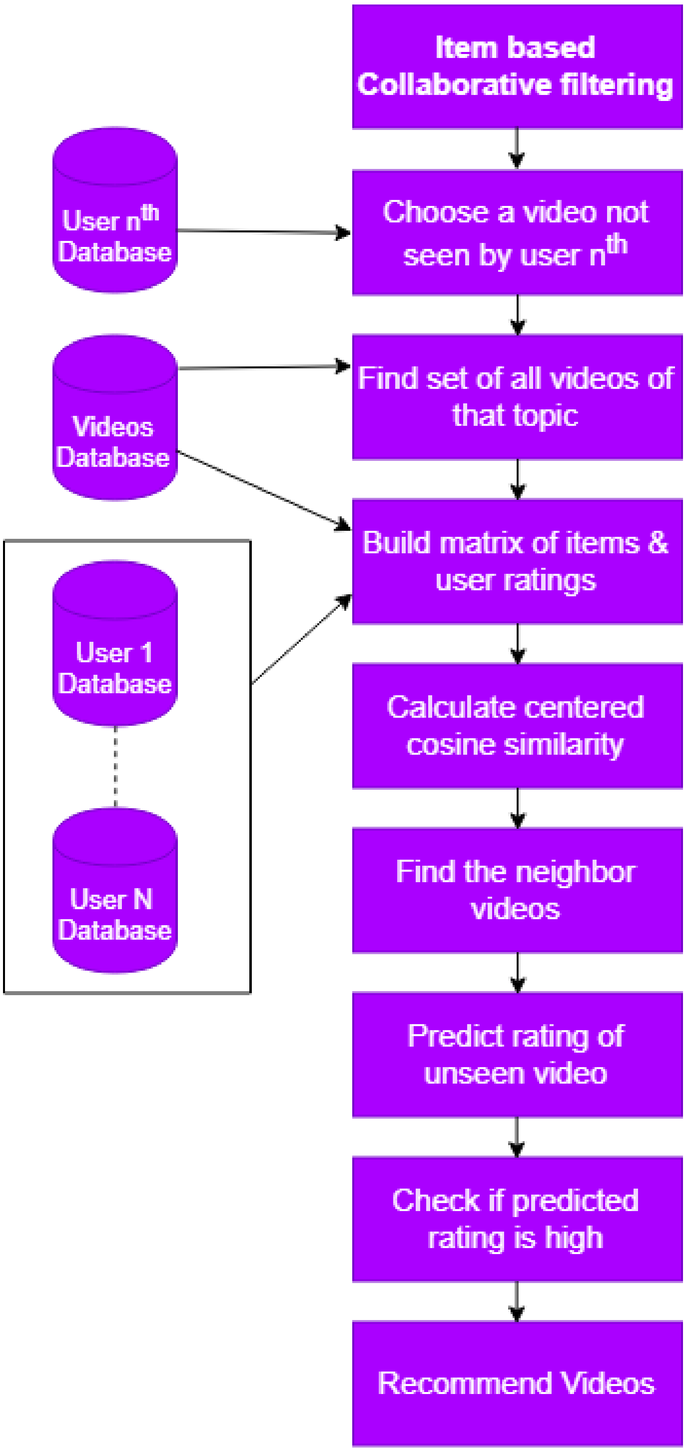

Figure 5.3: *Workflow of item-based collaborative filtering method*

Chapter 6

Result Analysis

6.1 Evaluation Process of the System

The model of the system are assessed or evaluated using various measures and calculations. All of these terms are described thoroughly in this section.

6.1.1 Confusion Matrix

According to Sunasra [22], a confusion matrix is used widely for evaluating the accuracy and performance of the model. It is used where the output is classified into two or more classes. The table of the confusion matrix is divided into two dimensions of 'Actual' and 'Predicted' and sets of classes in both dimensions. The columns are the 'Actual' classifications and the rows are the 'Predicted' classifications.

Terms associated with the confusion matrix:

- TP – True positives are the cases where the actual class was true and predicted is also true.
- TN – True negatives are the cases where the actual class was false and predicted is also false.
- FP – False positives are the cases where the actual class was false but predicted is true.
- FN – False negatives are the cases where the actual class was true but predicted is false.

Table 6.1: *Confusion Matrix*

		Actual	
		Positives	Negatives
Predicted	Positives	TP ($True\ Positives$)	FP ($False\ Positives$)
	Negatives	FN ($False\ Negatives$)	TN ($True\ Negatives$)

6.1.2 Accuracy

Accuracy is a measurement which is the ratio of number of correct predictions made by the model over all the predictions made. Accuracy is calculated using the given formula (6.1):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{6.1}$$

6.1.3 Precision

Precision is a measure that tells us, what proportion of our prediction was actually correct out of all the classes. Precision is calculated using the given formula (6.2):

$$Precision = \frac{TP}{TP + FP} \tag{6.2}$$

6.1.4 Recall

Recall or sensitivity is a measurement of out of all the positive classes, what proportion of correct cases were predicted as true cases. It is calculated using the given formula (6.3):

$$Recall = \frac{TP}{TP + FN} \tag{6.3}$$

6.1.5 Specificity

Specificity is a measure that reveals what proportion of the cases were actually false and also the prediction of the cases were false too. It is calculated using the given formula (6.4):

$$Recall = \frac{TN}{TN + FP} \tag{6.4}$$

6.1.6 F1 Score

F1 score is used to make it easier to measure recall and precision at the same time. It uses Harmonic Mean in place of Arithmetic Mean. It is calculated using the given formula (6.5):

$$F1\ Score = \frac{2 * Precision * Recall}{Precision + Recall} \tag{6.5}$$

6.2 Performance Evaluation of Used Algorithms

In the previous section, all essential measures and their formulae have been described. This section contains the results of our system after applying those measurement criteria.

6.2.1 Confusion Matrix of Content Based Filtering Method

Table 6.2: *Table for Confusion Matrix of Content Based Filtering Method*

$N = 200$

		Actual	
		Positives	Negatives
Predicted	Positives	66	0
	Negatives	24	110

6.2.2 Confusion Matrix of User-Based Collaborative Filtering Method

Table 6.3: *Table for Confusion Matrix of User-Based Collaborative Filtering Method*

$N = 150$

		Actual	
		Positives	Negatives
Predicted	Positives	105	0
	Negatives	30	15

6.2.3 Confusion Matrix of Item-Based Collaborative Filtering Method

Table 6.4: *Table for Confusion Matrix of Item-Based Collaborative Filtering Method*

$N = 200$

		Actual	
		Positives	Negatives
Predicted	Positives	60	0
	Negatives	40	100

6.2.4 Performance Measurement for Used Algorithms

The table shows the percentage of accuracy, precision, recall, specificity and F1 score of content based filtering method, user-based collaborative filtering method and item-based collaborative filtering method.

Table 6.5: *Overall Performance Measurement*

Details	**Content Based Filtering Method**	**User-Based Collaborative Filtering Method**	**Item-Based Collaborative Filtering Method**
Accuracy	88.0%	80.0%	80.0%
Precision	100.0%	100.0%	100.0%
Recall	73.0%	77.8%	60.0%
Specificity	100.0%	100.0%	100.0%
F1 Score	84.39%	87.5%	75.0%

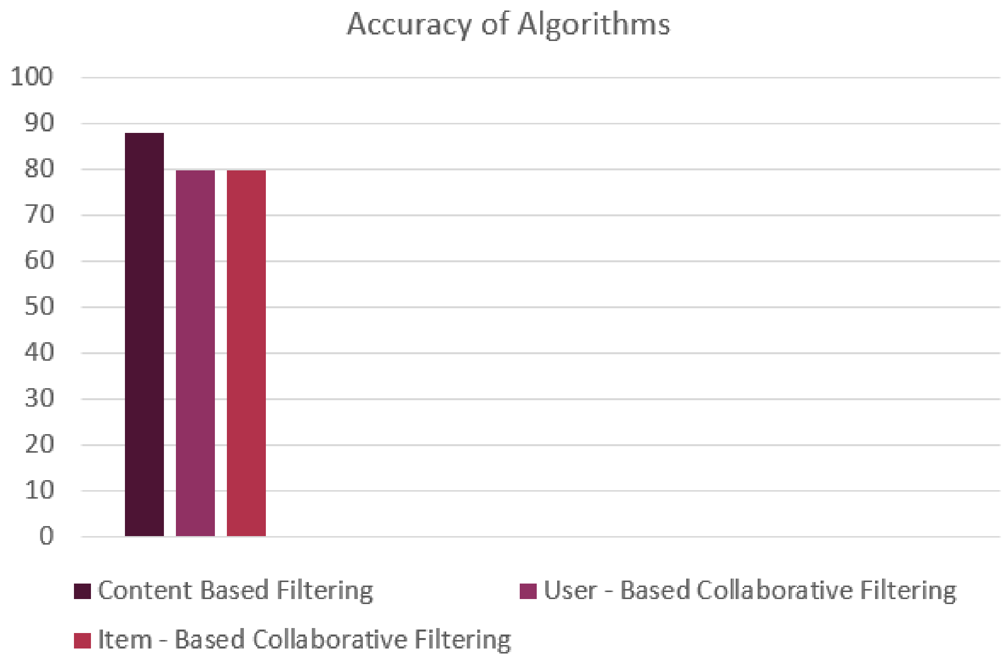

Figure 6.1: *Bar chart of Accuracy for the applied algorithms*

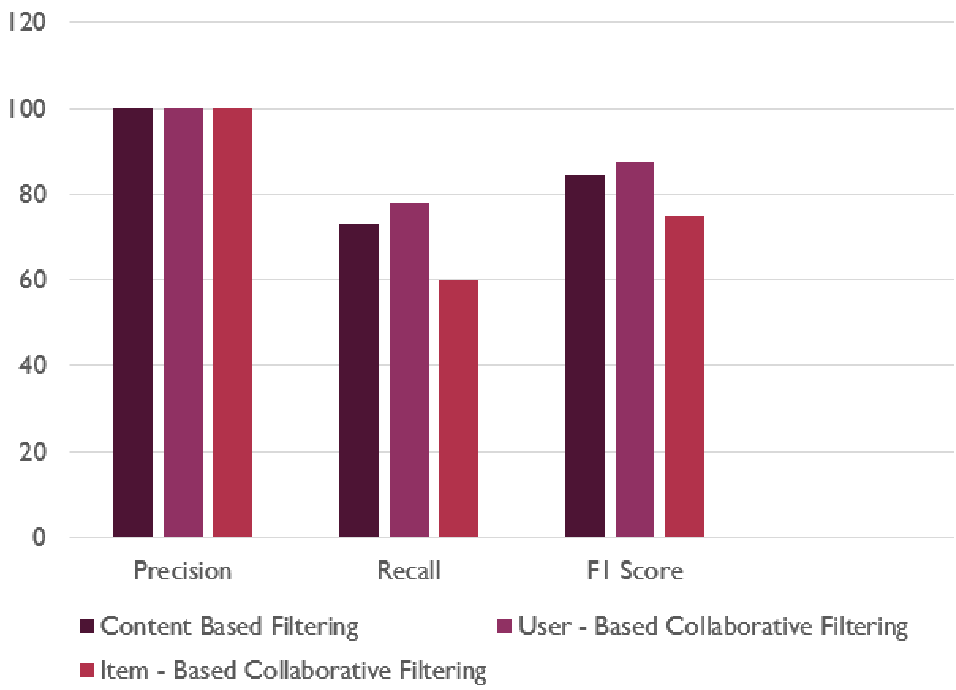

Figure 6.2: *Bar chart of Precision, Recall and F1 Score for the applied algorithms*

6.3 Comparison

We have designed an e-learning video recommendation system for CS undergraduate students which imports video tutorials' links from YouTube along the videos' necessary parameters. Unfortunately, we could not find enough past research papers on this kind of video recommendation system that is correctly relevant to our research. That is why, we could not compare our results with existing researches' results. From our analysis, precision, recall, specificity and F1 score – values of all these measures for the three filtering methods are similar. Moreover, accuracy score is the foremost measurement used widely to differentiate the performance or usefulness of any model. The accuracy of the content based filtering method is 88% whereas both the user-based collaborative filtering and the item-based collaborative filtering have accuracy values of 80%. Considering the accuracy, it can be deduced that the content based filtering method yields most accurate results so it has the best performance among these three. Content based filtering method recommends items considering the user's past history and seen or rated videos. So, in our case, the content based filtering method will not recommend anything to a fresher student who has not yet seen any videos. Whereas, the other two algorithms would still be able to recommend some videos to the fresher. Moreover, the accuracy of both user-based collaborative filtering and item-based collaborative filtering are 80% which are not so bad than that of content based filtering. In addition, the overall performance and correctness of all three algorithms are similar when executed on our datasets. To conclude, observing and measuring all the aspects, we used all three methods to form a hybrid approach for our recommendation system where each and every user is recommended at least something.

Chapter 7

Future Works and Conclusion

7.1 Future Work Plan

In future, we hope to collaborate with a web-developer team and built a Web Application using the design of our system done for this research paper. It will be an e-learning based web application named 'Learn Online'. The users are the students who will register on the application when using it for the first time. Later, they can sign in with their email address and password. The registration form will need the students to enter some information about themselves.

After logging in, the user will be displayed his home page. Here, user can search for new video tutorials, view his course codes or titles, playlist of saved videos and view a list of recommended videos for him.

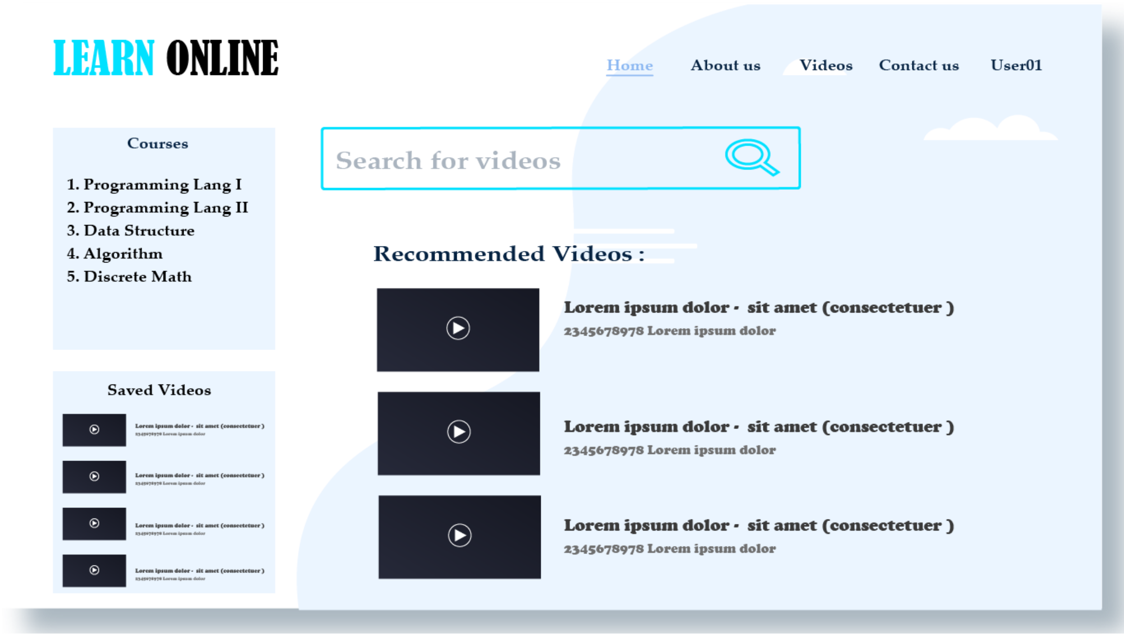

Figure 7.1: *Home page of the web application for User01*

When the user searches for new videos, he has to type the keywords or title for the video he is looking for. This search will be executed on YouTube and the necessary information about that video will be extracted and pulled to our web application using web crawler. A program or automated script which browses the World Wide Web in a methodical, automated manner is called a web crawler (also known as a web spider or web robot). This process is called Web crawling or spidering. Web crawlers are able to collect various information like - the URL of the website, the Web page content, the links in the webpage and the destinations leading from those links, the web page title and any other relevant information. All the necessary collected information of the video which might be required for the filtering algorithms to provide future recommendations to the user will then be stored in the system's database.

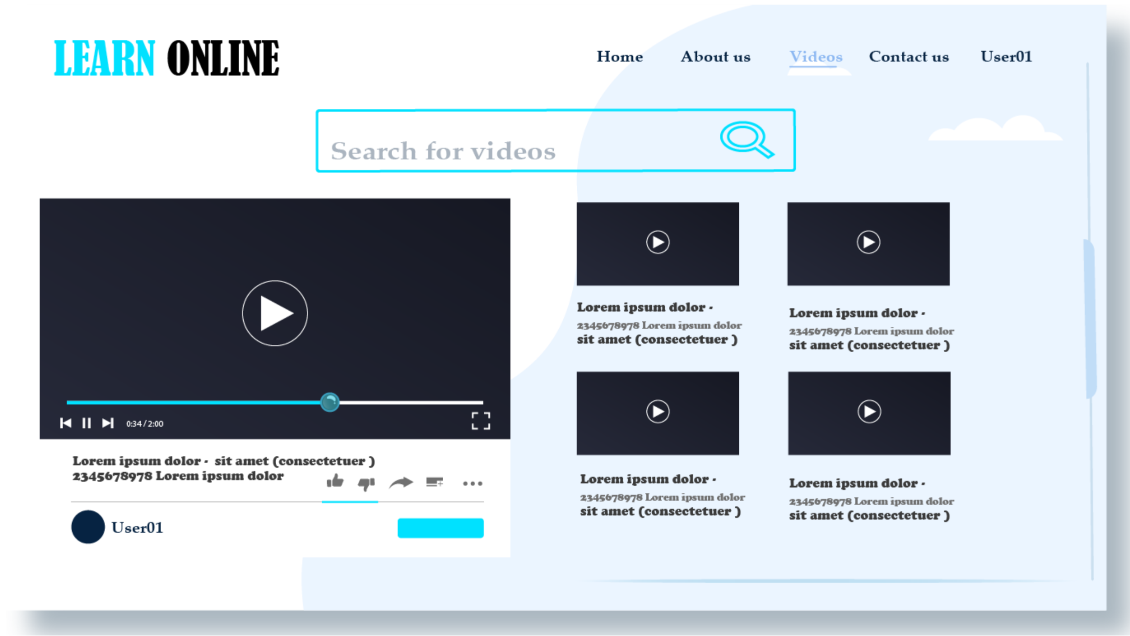

Figure 7.2: *View of how a video is displayed on the web application for User01*

7.2 Conclusion

At present, life has become a race, the faster one moves the more accomplishments he can achieve in life. People want to learn and attain skills at a faster pace and in an easier way. That is why, E-learning is growing more popular as days are passing by. Online learning saves the energy and time of going to institutions for acquiring knowledge as anybody can learn from anywhere through this process of learning. E-learning also saves money of enrolling for courses as existing various online learning sources are absolutely cost free. Our future plan is to build the web application 'Learn Online' to make future students' lives easier using the strategies narrated in this research paper. In this research paper, after analyzing the results of our research, we have proposed a hybrid filtering approach to build a video recommendation system for e-learning that includes content based filtering, user-based collaborative filtering and item-based collaborative filtering. Hopefully, we will be able to effectively help students to successfully study for their better future.

Bibliography

[1] O. R. Zaiane, "Building a recommender agent for e-learning systems", vol. 1, pp. 55–59, Dec. 2002. DOI: 10.1109/CIE.2002.1185862.

[2] J. Lu, "A personalized e-learning material recommender system", pp. 23–28, Jan. 2004.

[3] C.-M. Chen, H.-M. Lee, and Y.-H. Chen, "Personalized e-learning system using item response theory.", *Computers Education*, vol. 44, pp. 237–255, Apr. 2005. DOI: 10.1016/j.compedu.2004.01.006.

[4] T. Y. Tang and G. McCalla, "Smart recommendation for an evolving e-learning system: Architecture and experiment", *International Journal on E-learning*, vol. 4, pp. 105–129, 2005.

[5] N. Soonthornphisaj, E. Rojsattarat, and S. Yim-ngam, "Smart e-learning using recommender system", *Lecture Notes in Computer Science*, vol. 4114, pp. 518–523, Aug. 2006. DOI: 10.1007/11816171_63.

[6] K. Khribi, M. Jemni, and O. Nasraoui, "Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval", in *2008 Eighth IEEE International Conference on Advanced Learning Technologies*, vol. 12, IEEE, Aug. 2008, pp. 241–245, ISBN: 978-0-7695-3167-0. DOI: 10.1109/ICALT.2008.198.

[7] H. Tan, J. Guo, and Y. Li, "E-learning recommendation system", in *International Conference on Computer Science and Software Engineering*, vol. 5, IEEE, Dec. 2008, pp. 430–433.

[8] J. Bobadilla, F. Serradilla, and A. Hernando, "Collaborative filtering adapted to recommender systems of e-learning", *Knowledge-Based Systems*, vol. 22, pp. 261–265, May 2009. DOI: 10.1016/j.knosys.2009.01.008.

[9] K. I. B. Ghauth and N. A. Abdullah, "Automatic recommendations for e-learning personalization based on web usage mining techniques and information retrieval", in *2009 Ninth IEEE International Conference on Advanced Learning Technologies*, IEEE, Jul. 2009, pp. 194–196.

[10] L. Zhuhadar, O. Nasraoui, R. Wyatt, and E. Romero, "Multi-model ontology-based hybrid recommender system in e-learning domain", in *2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology*, vol. 3, IEEE, Sep. 2009, pp. 91–95. DOI: 10.1109/WI-IAT.2009.238.

[11] K. I. Ghauth and N. A. Abdullah, "Measuring learner's performance in e-learning recommender systems", *Australasian Journal of Educational Technology*, vol. 26, no. 6, pp. 764–774, 2010.

[12] R. Sikka, A. Dhankhar, and C. Rana, “A survey paper on e-learning recommender system”, *International Journal of Computer Applications*, vol. 47, pp. 27–30, Jun. 2012. DOI: 10.5120/7218-0024.

[13] K. Verbert, N. Manouselis, X. Ochoa, M. Wolpers, H. Drachsler, I. Bosnic, and e. duval erik, “Context-aware recommender systems for learning: A survey and future challenges”, *IEEE Transactions on Learning Technologies*, vol. 5, pp. 318–335, Oct. 2012. DOI: 10.1109/TLT.2012.11.

[14] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of massive dataset*, 2^{nd}. Cambridge university press, 2014.

[15] S. Das, “Beginners guide to learn about content based recommender engines”, Aug. 2015, [online] https://www.analyticsvidhya.com/blog/2015/08/beginners-guide-learn-content-based-recommender-systems/?fbclid=IwAR1i25kkra9ub82zXW [Online]. Available: https://www.analyticsvidhya.com/blog/2015/08/beginners-guide-learn-content-based-recommender-systems/?fbclid=IwAR1i25kkra9ub82zXW

[16] T. Hopmans, “A recommendation system for blogs: Setting up the prerequisites (part 1)”, Aug. 2015, [online] https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13/. [Online]. Available: https://www.themarketingtechnologist.co/building-a-recommendation-engine-for-geek-setting-up-the-prerequisites-13/.

[17] F. E. Gunawan, A. Maryanto, Y. Udjaja, S. Candra, and B. Soewito, “Improvement of e-learning quality by means of a recommendation system”, in *2016 11th International Conference on Knowledge, Information and Creativity Support Systems (KICSS)*, Nov. 2016, pp. 1–4. DOI: 10.1109/KICSS.2016.7951421.

[18] P. Dangeti, *Statistics for machine learning*. Packt Publishing Ltd, 2017.

[19] M. Maravanyika, N. Dlodlo, and N. Jere, “An adaptive recommender-system based framework for personalised teaching and learning on e-learning platforms”, in *2017 IST-Africa Week Conference (IST-Africa)*, May 2017, pp. 1–9. DOI: 10.23919/ISTAFRICA.2017.8102297.

[20] C. Pinela, “Recommender systems—user-based and item-based collaborative filtering”, Nov. 2017, [online] https://medium.com/@cfpinela/recommender-systems-user-based-and-item-based-collaborative-filtering-5d5f375a127f?fbclid=IwAR0sozWqafurWpui2nczLizliL3aRc9xeWfE0OtPkpQ7z1E2NNn2wiwCAaU. [Online]. Available: https://medium.com/@cfpinela/recommender-systems-user-based-and-item-based-collaborative-filtering-5d5f375a127f?fbclid=IwAR0sozWqafurWpui2nczLizliL3aRc9xeWfE0OtPkpQ7z1E2NNn2wiwCAaU.

[21] J. Silge and D. Robinson, *Text mining with R: A tidy approach.* ” O’Reilly Media, Inc.”, 2017.

[22] M. Sunasra, “Performance metrics for classification problems in machine learning”, Nov. 2017, [online] https://medium.com/thalus-ai/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b?fbclid=IwAR36h1zOWgBkoPstK9bQGXWh3F-OnbzAf_CBS7qpZhOWwK4l-sfaQIMaAbU. [Online]. Available: https://medium.com/thalus-ai/performance-metrics-for-classification-problems-in-machine-learning-part-i-b085d432082b?fbclid=IwAR36h1zOWgBkoPstK9bQGXWh3F-OnbzAf_CBS7qpZhOWwK4l-sfaQIMaAbU.

[23] J. Tarus, Z. Niu, and G. Mustafa, "Knowledge-based recommendation: A review of ontology-based recommender systems for e-learning", *Artificial Intelligence Review*, vol. 50, pp. 21–48, Jan. 2017. DOI: 10.1007/s10462-017-9539-5.

[24] A. Tomar, "Item-item collaborative filtering recommender system in python", Oct. 2017, [online] https://medium.com/@tomar.ankur287/item-item-collaborative-filtering-recommender-system-in-python-cf3c945fae1e?fbclid=IwAR2hsZfG9hKmv8-gDKz_4vZRuA_bTcYNahcb3QFJH4iAe7GUnH2yzCjImRc. [Online]. Available: https://medium.com/@tomar.ankur287/item-item-collaborative-filtering-recommender-system-in-python-cf3c945fae1e?fbclid=IwAR2hsZfG9hKmv8-gDKz_4vZRuA_bTcYNahcb3QFJH4iAe7GUnH2yzCjImRc.

[25] N. Albatayneh, K. Imran, and F.-F. Chua, "Utilizing learners' negative ratings in semantic content-based recommender system for e-learning forum", *Educational Technology Society*, vol. 21, pp. 112–125, Jan. 2018.

[26] M. A. Hasan and D. Schwartz, "Recadvisor: Criteria-based ph. d. supervisor recommendation", in *The 41st International ACM SIGIR Conference on Research Development in Information Retrieval*, vol. 12, ACM, Jun. 2018, pp. 1325–1328, ISBN: 978-0-9981331-2-6. DOI: http://hdl.handle.net/10125/59622.

[27] S. Huang, "Introduction to recommender system. part 1 (collaborative filtering, singular value decomposition)", Jan. 2018, [online] https://hackernoon.com/introduction-to-recommender-system-part-1-collaborative-filtering-singular-value-decomposition-44c9659c5e75?fbclid. [Online]. Available: https://hackernoon.com/introduction-to-recommender-system-part-1-collaborative-filtering-singular-value-decomposition-44c9659c5e75?fbclid.

[28] A. Klašnja-Milićević, M. Ivanović, B. Vesin, and Z. Budimac, "Enhancing e-learning systems with personalized recommendation based on collaborative tagging techniques", *Applied Intelligence*, vol. 48, no. 6, pp. 1519–1535, Jun. 2018, ISSN: 1573-7497. DOI: 10.1007/s10489-017-1051-8. [Online]. Available: https://doi.org/10.1007/s10489-017-1051-8.

[29] D. Lew, B. Sowell, and L. E. Steinberg, *Recommender systems.* Carleton College, 2018.

[30] S. Luo, "Introduction to recommender system", Dec. 2018, [online] https://towardsdatascience.com/intro-to-recommender-system-collaborative-filtering-64a238194a26?fbclid=IwAR3HW6kQEiA9Eqvzy0OH0TUOexG6RSmJ-Qyxoad2LeEX-ssiBECPYnHSxak.

[31] K. Sandhu and H. Alharbi, "Explaining and predicting continuance usage intention of e-learning recommender systems: An empirical evidence from saudi arabia", *International Journal of Business Information Systems*, vol. 29, pp. 297–323, Jan. 2018. DOI: 10.1504/IJBIS.2018.10009149.

[32] A. Sheshasaayee and M. Nazreen Bee, "E-learning: Mode to improve the quality of educational system", pp. 559–566, Oct. 2018. DOI: 10.1007/978-981-10-5547-8_58.

[33] R. Verma, "Applying predective analytics in elective course recommender system while preserving student course preferences", pp. 1–38, Jun. 2018.

[34] G. Rodríguez, “Introduction to recommender systems in 2019”, 2019, [online] https://tryolabs.com/blog/introduction-to-recommender-systems/?fbclid=IwAR1hjEPhig79Jm38Z5Butuq2EhPclNIUiqVwZrpYwgVCk4BV0DtSqaAIRes. [Online]. Available: https://tryolabs.com/blog/introduction-to-recommender-systems/?fbclid=IwAR1hjEPhig79Jm38Z5Butuq2EhPclNIUiqVwZrpYwgVCk4BV0DtSqaAIR