TRIBHUWAN UNIVERSITY

INSTITUTE OF

ENGINEERING

PURWANCHAL CAMPUS,

DHARAN-8

A MINOR PROJECT PROPOSAL ON ONLINE COURSE RECOMMENDATION SYSTEM

SUBMITTED BY

OCEAN SITAULA (PUR077BCT053)
PRABESH SHARMA (PUR077BCT054)
SHAILESH DEVKOTA (PUR077BCT077)
SUCHITA KUMARI SAH (PUR077BCT082)

SUBMITTED TO

DEPARTMENT OF ELECTRONICS AND COMPUTER ENGINEERING

IOE - PURWANCHAL CAMPUS, DHARAN-8

SUBMISSION DATE

2080 / 09 / 19

ACKNOWLEDGEMENT

I express my sincere appreciation to the Head of the Department (HOD), Mr. Pravin Sangroula and the Deputy Head of the Department, Mr. Pukar Karki for their invaluable guidance, insightful suggestions,constructive feedback, and continuous support throughout the preparation of this proposal.

I am deeply thankful to the esteemed teachers and faculty members whose valuable feedback, encouragement, and scholarly insights have enriched this proposal. Their dedication to academic excellence and commitment to fostering knowledge have been a source of inspiration. Their collective guidance have immensely contributed to the refinement and improvement of this proposal.

TABLE OF CONTENTS

1. INTRODUCTION

1.1 Background Study

A recommendation system (sometimes replacing 'system' with a synonym such as platform or engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. They are primarily used in commercial applications. Recommendation systems are utilized in a variety of areas and are most commonly recognized as playlist generators for video and music services like Netflix, YouTube and Spotify, product recommendation for services such as Amazon, or content recommendation for social media platforms such as Facebook and Twitter. These systems can operate using a single input, like music, or multiple inputs within and across platforms like news, books and search queries. There are also popular recommendation systems for specific topics like restaurants and online dating.
Recommendation systems have also been developed to explore research articles and experts, collaborators and financial services.

1.2 Problem Definition

The need of user to be recommended courses being based on popularity or similarity of the course also considering the past-viewing rating of other users.

1.3 Objectives

1.3.1 To recommend the best courses based on user preference.

1.3.2 To compile all the courses from various MOOC platforms into a single website.

1.4 Scope of Project

It involves data collection, content analysis, and algorithm implementation for personalized course suggestions. The system aims to enhance user engagement and satisfaction within the platform through accurate and tailored recommendations.

2. LITERATURE REVIEW

2.1. Historical Evolution

Early Collaborative Filtering Based Recommendation Systems were introduced during 1990s [3] which laid the foundation for user-item interactions to make recommendations, which later influenced online course recommendation systems. Later, during the early 2000s, modern approaches to Collaborative filtering, Content Based Filtering and Hybrid Recommendation Models were introduced. As several studies [4, 5] suggest, the evolution of machine learning led to better approaches such as Matrix Factorization and Reinforcement Learning which greatly improved the efficiency of the Recommendation Systems.

2.2 Algorithms and Models

2.2.1 Collaborative Filtering

It includes user-based and item-based approaches, where user similarities or item similarities are computed to generate recommendations. Moreover, to assess the effect of course dependency on recommendations, the SPADE algorithm was applied on course sequences as shown by [3].

2.2.2 Content-Based Filtering

It focuses on analyzing item (course) attributes and user preferences. As suggested in [2] recommendations can be made based on the similarity between item characteristics and user profiles.

2.2.3 Matrix Factorization

Matrix factorization methods, such as Singular Value Decomposition (SVD) and Alternating Least Squares (ALS), decompose user-item interaction matrices to discover latent factors influencing user preferences.

2.2.4 Hybrid Model

Hybrid models combine multiple recommendation techniques, such as collaborative filtering and content-based filtering, which according to the report [1] provides more accurate and diverse recommendations.

2.3 Data Pre-Processing and Feature Engineering

Data processing and feature engineering play a crucial role in refining data and enhancing recommendation accuracy in online courses. These steps involve cleaning data neatly, organizing user actions and understanding user interactions with course content using language and images as seen in [2]. These methods also include the extraction of meaningful information from course content and user behavior. By combining different techniques, they refine data and contribute to the system's ability to provide better, personalized course suggestions to users.

2.4 Gaps and Challenges

Recommendation systems for MOOCs play a significant role in reducing dropout rates and enhancing learner's success. However, there are still many gaps and challenges to implement a successful recommendation system because an effort hasn't been made yet for the collection of adequate datasets according to [5].

The online course recommendation system till the date have been unable to compile the courses of various websites into a single platform efficiently to build a better MOOC recommendation system than the ones that have been built before.

3. METHODOLOGY

3.1 System Design

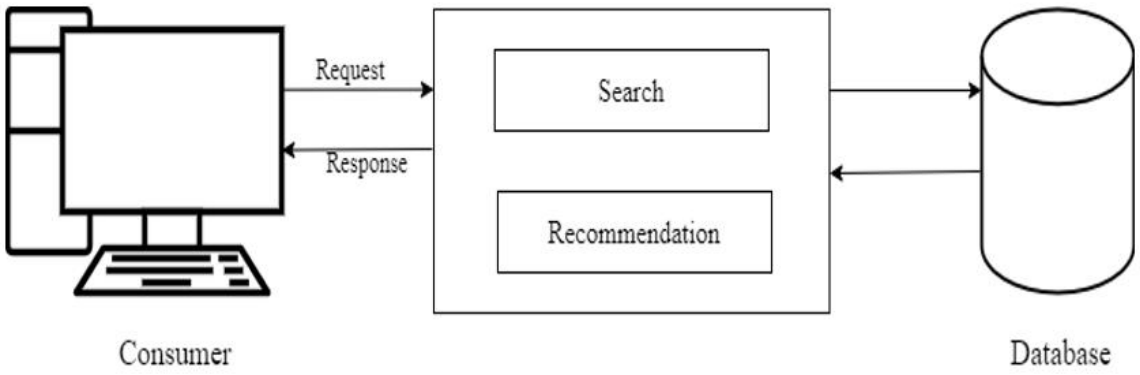


Fig. 1 : System Design

3.2 Proposed Block Diagram

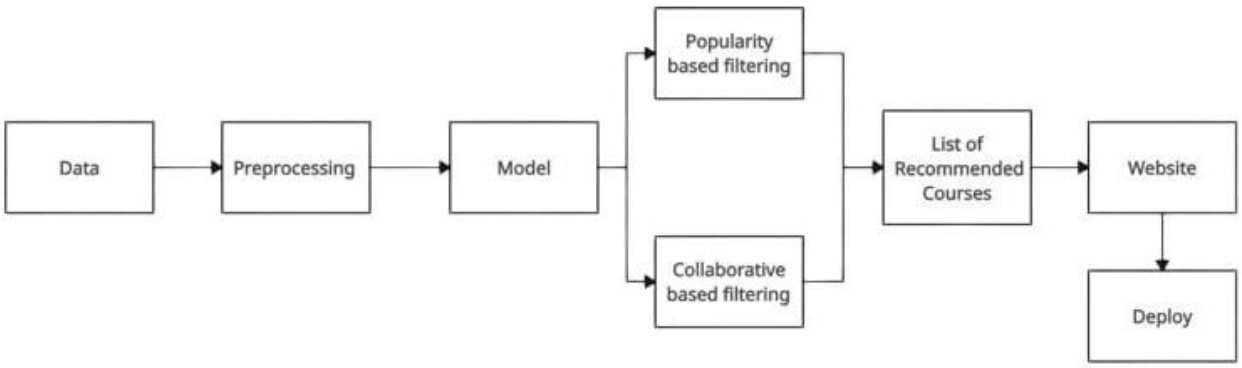


Fig. 2 : Proposed Block Diagram

3.3 Tools and Techniques

3.3.1 Tools

- **Python**

Python, when used in machine learning, offers developers of all skill sets exceptional versatility and power. Developers can use Python to develop a variety of applications because it integrates well with other software while its simple syntax

makes it a good choice for coding algorithms and collaborating across teams.

- **Python Libraries**

  I. Django

  Django is a Python-based web application framework that is free and open source. They're grouped together and allow you to build apps or websites from scratch rather than starting from scratch.

  II. Numpy

  NumPy can used for working with arrays. It also has functions for working in domain of linear algebra, Fourier transform, and matrices.

  III. Pandas

  Pandas can be useful tool for working with data sets. It has functions for analyzing, cleaning, exploring, and manipulating data.

  IV. Scikit-learn

  Scikit-learn is an open source machine learning library that supports supervised and unsupervised learning. It also provides various tools for model fitting, data pre-processing, model selection, model evaluation and many other utilities.

- **HTML**

HTML (Hyper Text Markup Language) is the code that is used to structure a web page and its content.

- **CSS**

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

- **JavaScript**

JavaScript (JS) is a lightweight interpreted programming language with first-class functions. It is most well-known as the scripting language for Web pages.

3.3.2 Techniques

i. Requirement/Data Gathering: Data is the most important and foundation for machine learning projects. Gathering data from various datasets is key for a recommendation system. The more the data available the better the recommending results.

ii. Pre-processing: In the pre-processing stage, filtering and making ready the data for the project, we will make some changes such as, we will build tags that will describe the data and help us to calculate its similarity with other data.

iii. System designing: In this system design phase, we design the system which is easily understood by the end-user i.e., user friendly. We design data flow diagrams to understand the system flow and

system module and sequence of execution.

iv. Model implementation and testing: We will create a framework of data that will coordinate with the code this phase involves the core part of our project that is coding and model designing. The model will make sure the project works well at the local level. The different test cases are performed to test whether the project module is giving the expected outcome in the assumed time.

v. Website designing: After we have created a working model, we will create the same into a website. This stage will involve designing an immersive UI.

vi. Deployment of System: Once the functional and non-functional testing is done, the product is deployed in the virtual environment or released into the local hosting like Heroku, over the internet.

3.4. Data Source

3.4.1. Primary Source

Primary data refers to the first hand data gathered by the researchers themselves using observations, experiments, surveys, questionnaire, personal interview, etc. It is reliable as real time data is used instead of already available past data sets.

For our project, the datasets has been collected manually through various MOOC platforms. Our data set consists of the following attributes:

- Course Id (Numerical Data)
- Course Name (Text Data)
- Source (Text Data)
- URL (Text Data)
- Is Paid (Categorical Data)
- Level (Categorical Data)
- Number of Enrollments (Numerical Data)
- Duration (Numerical Data)
- Rating (Numerical Data)
- Review (Numerical Data)
- Published Year (Numerical Data)
- Genre (Categorical Data)

Minimum Number of Tuples expected : 1000

4. Implementation

The Proposed System makes use of different Algorithms and Methods for the implementation of our Recommendation System.

4.1 CountVectorizer:

CountVectorizer creates a matrix in which each unique word is represented by a column of the matrix, and each text sample from the document is a row in the matrix. The value of each cell is nothing but the count of the word in that particular text sample. This can be visualized as follows

	at	each	four	geek	geeks	geeksforgeeks	help	helps	many	one	other	two
document[0]	0	0	0	1	1	0	0	1	0	1	0	1
document[1]	0	0	1	0	2	0	1	0	0	0	0	1
document[2]	1	1	0	1	1	1	0	1	1	0	1	0

Fig. 3 : Visualization of CountVectorizer

4.2 TfidVectorizer:

It is a popular method for converting a collection of raw documents to a matrix of TF-IDF features. TF-IDF reflects the importance of a term within a document relative to its frequency across multiple documents. The vectorizer calculates a weight for each term in each document, and the resulting matrix can be used as input for various machine learning algorithms.

Formula:

$$TF(t, d) = \frac{\text{number of times t appears in d}}{\text{total number of terms in d}}$$

$$IDF(t) = \log \frac{N}{1+df}$$

$$TF - IDF(t, d) = TF(t, d) * IDF(t)$$

One of the techniques (CountVectorizer or TdifVectorizer) will be selected for creating sparse matrix considering maximum accuracy for our model.

4.3 Cosine Similarity:

Cosine similarity is a measure of similarity between two non-zero vectors of an inner product space that measures the cosine of the angle between them.

Formula:

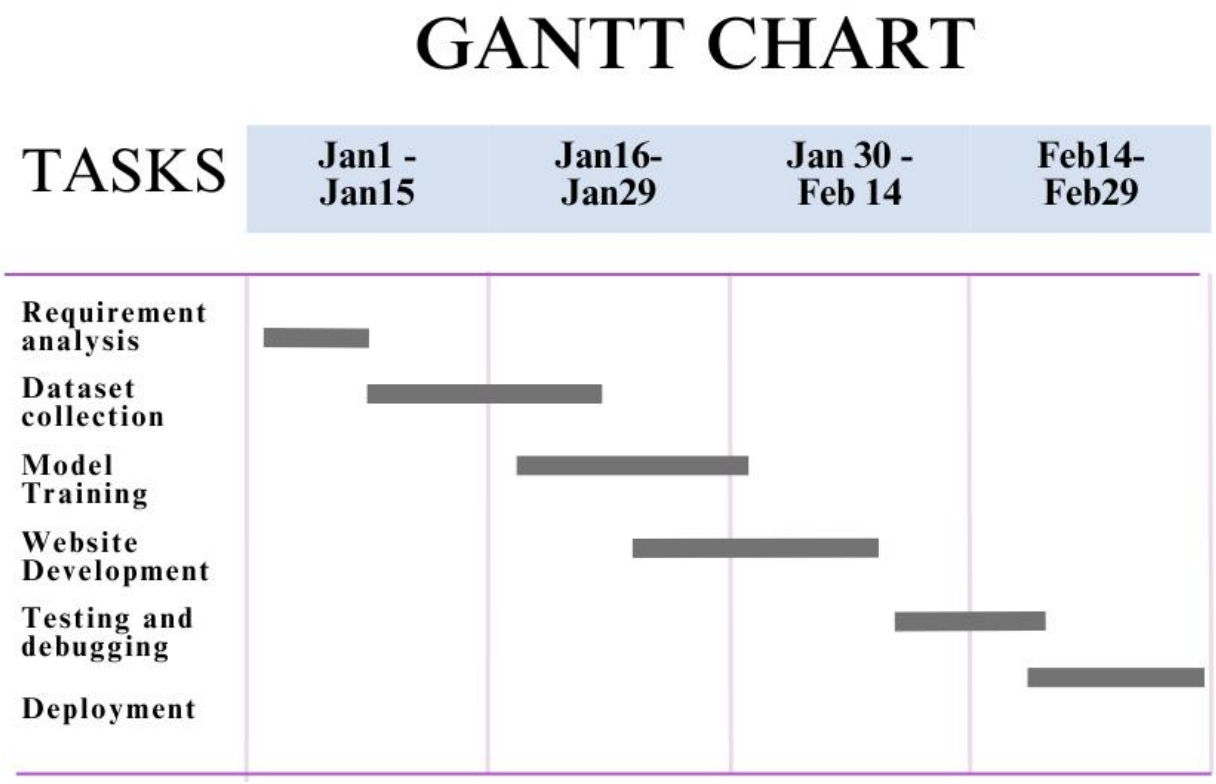
$$\cos\theta = \frac{\vec{a} * \vec{b}}{\| \vec{a} \| \| \vec{b} \|} = \frac{\sum_1^n a_i b_i}{\sqrt{\sum_1^n a_i^2} \sqrt{\sum_1^n b_i^2}}$$

Where $\vec{a}.\vec{b} = \sum_1^n a_i b_i = a_1 b_1 + a_2 b_2 ... + a_n b_n$ is the dot product of two vectors.

5. EXPECTED OUTPUT

The goal of the system is to provide recommendations to help users find courses of their interests either on the basis of popularity or similarity. The system is expected to produce at least 85% of accuracy.

6 . EXPECTED SCHEDULE



REFERENCES

[1] S.B. Aher and L.M.R.J. Lobo, "Course Recommender System in E-learning," International Journal of Computer Science and Communication, Jan 2012.

[2] D. Debashis, L. Sahoo, and S. Dutta, "A survey on recommendation system," International Journal of Computer Applications, 2017.

[3] O. Raghad, R. Duwairi, and A.A. Aiad. "A collaborative recommendation system for online courses recommendations," International conference on deep learning and machine learning in emerging applications (Deep-ML), 2019.

[4] I. Uddin, A.S. Imran, K. Muhammad, N. Fayyaz and M. Sajjad," A systematic mapping review on MOOC recommender systems," IEEE Access, Jul 29, 2021.

[5] J. W. Tzeng, N. F. Huang, A. C. Chuang, T.W. Huang and H. Y. Chang, "Massive open online course recommendation system based on a reinforcement learning algorithm," Neural Computing and Applications, Jun 17, 2023.