Course Recommendation System

Introduction

Today's learners have access to a wide range of courses in various fields, thanks to online platforms such as Udemy, Coursera, and others. However, the real challenge lies in choosing the course that best suits an individual's skills, interests, and level. To enhance the learning experience, personalized recommendations will be provided to everyone based on an intelligent system developed through machine learning and data-driven analysis, saving learners time and effort.

Problem Statement

Most educational platforms still rely on manual searches or general classifications such as "most popular," "highest-rated," or "newest." However, these methods fail to consider the diverse interests, backgrounds, and learning goals of individual users as a result, learners often spend excessive time searching for suitable courses, which can lead to frustration and a decline in engagement. Moreover, the lack of intelligent recommendation systems limits the platform's ability to provide personalized learning experiences. Addressing this problem is crucial to enhancing user satisfaction, improving learning outcomes, and increasing course participation rates.

Objectives

1. To collect and process a dataset containing detailed information about courses, including titles, educational levels, target audiences, and ratings.
2. To identify and analyze the key factors that influence users' decisions when selecting a course, based on their interests and learning preferences.
3. To design and implement a content-based recommendation model using similarity metrics.
4. To evaluate the accuracy, relevance, and effectiveness of the recommendations generated by the model.
5. To develop a simple and user-friendly interface that clearly presents the recommended courses to users.

Proposed Methodology /Approach

- **Data Collection and Processing**
  Using a publicly available dataset from e-learning platforms, such as Udemy or Coursera course data, the data will be downloaded as a CSV or JSON file. From Kaggle or another platform, the data will be cleaned and processed by addressing missing values, preparing the text for analysis, and removing duplicates. The text is then preprocessed. Finally, the data will be prepared for the model (Feature preparation), where each course is represented as a vector representing its meaning. These vectors are used to calculate a similarity score between courses based on their content.

- **Feature Engineering**
  Applying text processing techniques to convert text into a numerical representation that can be mathematically analyzed. Texts will be cleaned using Python libraries. Then, the text will be segmented to facilitate analysis (Tokenization). Words will then be converted to their original form to reduce redundancy (Lemmatization). Then, the text will be converted to numbers (Vectorization). Similarity calculations will be performed.

- **Model Implementation**
  Use a content-based filtering model with a similarity metric to discover and recommend similar courses to the user based on the input they have entered.

- **Interface Design**
  Develop simple user-friendly interfaces using Django or Flask as a web-based UI to allow users to enter their interests or preferred topics and view the recommended courses.

- **Model Evaluation**
  Evaluating the performance and efficiency of the system by the user's feedback or by appropriate evaluation metrics such as accuracy, precision, and ranking effectiveness to measure how well the model provides relevant recommendations.

Expected Outcomes

The expected outcome of this project is the development of an effective course recommendation system with an intuitive interface that suggests suitable courses to users based on their individual interests and preferences. This system aims to help learners quickly identify relevant courses without the need for lengthy manual searching.

Expected outcomes also include:

- Additional benefits for educational platforms include increased course enrollment rates resulting from recommendations.
- A clean, organized dataset containing detailed information about educational courses available on online platforms.
- An interactive, easy-to-use interface that enables users to enter their interests and easily browse the recommended courses.
- A content-based recommendation model that analyzes the course descriptions entered by the user and provides accurate recommendations.