Notebook 2: Case3.ipynb

1. Recommender System

Objective:

The goal of the recommender system is to suggest the top 5 subjects (e.g., AI, Data Science, etc.) that a student is most likely to succeed in, based on their past quiz performance, engagement data, and academic history.

Key Steps:

1. Data Preprocessing

- **Merging Multiple Data Sources**: The recommender system merges various datasets, such as student demographics (age, gender, major), engagement metrics (logins, video views, time spent on the platform), and quiz performance.
- **Standardization**: Numeric features like quiz scores, time spent, and logins are standardized to ensure that the recommender system treats all features on a similar scale. This step helps to remove biases caused by differences in the scales of data.

Why this matters:

- **Personalized Learning**: By combining multiple sources of student data, the recommender system can tailor suggestions based on a more complete picture of the student's behavior and academic history. This personalization is crucial in education, as different students have unique strengths and weaknesses.

2. Collaborative Filtering

- **Matrix Factorization**: The core technique used in the recommender system is **collaborative filtering** with **matrix factorization**. This method decomposes the student-subject interaction matrix into lower-dimensional matrices representing latent factors for both students and subjects.
  - **Latent Factors**: In this context, latent factors are the hidden features of students and subjects that explain the preferences (e.g., quiz performance in certain subjects, engagement with learning content).
  - **Student-Subject Matrix**: A matrix is created where each row represents a student, and each column represents a subject. The values in the matrix correspond to quiz performance or engagement level in that subject.

- **Matrix Decomposition**: The student-subject matrix is decomposed into two matrices: one for students and one for subjects. This allows the system to predict missing values in the matrix, which represent subjects the student has not interacted with yet.

Why this matters:

- **Learning Student Preferences**: Matrix factorization helps the recommender system understand hidden relationships between students and subjects. For example, a student may perform well in subjects that require logical thinking, so the system will recommend subjects with similar characteristics.

3. Generating Recommendations

- **Dot Product for Predictions**: Once the student and subject matrices are learned, the system computes the **dot product** of the two matrices to generate a predicted score for each subject that the student hasn't taken. The subjects with the highest predicted scores are recommended.
- **Top 5 Recommendations**: The system then selects the top 5 subjects for each student based on these predicted scores.

Why this matters:

- **Personalization**: The use of matrix factorization enables the system to generate personalized recommendations, ensuring that students are more likely to succeed in the subjects that are suggested.

4. Innovation:

- **Collaborative Filtering in Education**: While collaborative filtering is widely used in product recommendation (e.g., Netflix, Amazon), its application in education is innovative. It allows for personalized learning paths, helping students discover subjects where they are most likely to excel based on their historical performance and engagement metrics.
- **Subject Personalization**: Instead of generic recommendations, the system leverages both engagement and performance data to tailor subject recommendations. This approach can increase student success rates by recommending courses that match their strengths and interests.

---

2. Classification Model

Objective:

The classification model in this notebook aims to predict whether a student will successfully complete the recommended subjects based on their demographic data, engagement metrics, and past performance.

Key Steps:

1. Data Preprocessing

- **Data Merging**: As with the recommender system, the classification model merges student demographics, engagement data, and academic performance data into a single dataset.
- **Handling Missing Values**: Missing values in the dataset are imputed or removed to ensure that the model receives clean, complete data.
- **One-Hot Encoding**: Categorical features (e.g., subjects, gender, major) are converted into binary features using one-hot encoding. This allows the classification algorithm to handle non-numeric features.

Why this matters:

- **Feature Representation**: Properly preparing the data ensures that the machine learning model can learn effectively. For instance, one-hot encoding of categorical data allows the model to differentiate between students majoring in different subjects and how it impacts their completion likelihood.

2. Feature Engineering

- **Quiz Performance as a Key Feature**: Quiz scores, time spent, logins, and other engagement metrics are key features used to determine student performance and engagement.
- **Completion Score**: A custom completion score is calculated for each student based on the predicted probabilities of completing each subject. This score is a weighted combination of subject preferences and engagement data.

Why this matters:

- **Engagement-Based Prediction**: The model doesn’t just rely on past performance in terms of grades; it also considers how engaged students are in the platform (e.g., videos watched, logins, time spent). This is an innovative approach because it captures early indicators of whether a student will complete the subjects.

3. Model Training

- **Model Selection**: The classification model uses **XGBoost**, a powerful gradient boosting algorithm, to predict whether a student will complete all recommended subjects.
    - **XGBoost's Strengths**: XGBoost is particularly well-suited for this task because it can handle complex, high-dimensional data and is robust to missing data and noise. It also has strong performance with tabular data, which is common in educational datasets.
- **Training and Validation Split**: The data is split into training and validation sets to avoid overfitting and evaluate the model's generalization ability.

Why this matters:

- **Handling Complexity**: The use of XGBoost allows the model to handle the complexity of combining demographic, engagement, and performance data. Its ability to work with tabular data and complex relationships makes it an ideal choice for predicting student outcomes.

4. Evaluation Metrics

- **Accuracy, F1-Score, and Confusion Matrix**: The classification model is evaluated using metrics like accuracy, precision, recall, and the F1-score. A confusion matrix is also generated to understand how well the model is predicting both completions and non-completions.
    - **Accuracy**: Measures the percentage of correct predictions made by the model.
    - **F1-Score**: Provides a balance between precision and recall, useful for handling imbalanced datasets (e.g., more students failing to complete subjects than completing them).
- **Confusion Matrix**: Provides insight into false positives (predicting a student will complete but they don't) and false negatives (predicting a student won't complete but they do).

Why this matters:

- **Model Performance Understanding**: Evaluating the model using these metrics helps ensure that it not only predicts accurately but does so in a balanced manner, reducing the number of false positives and negatives.
- **Risk Identification**: Understanding where the model is making errors (e.g., predicting students will complete when they won't) allows educators to intervene more effectively by providing additional resources or support.

5. Feature Importance

- **XGBoost's Feature Importance**: The model's feature importance is calculated and visualized. Features like `quiz scores`, `logins`, and `time spent` are often the most influential in determining whether a student completes the recommended subjects.

Why this matters:

- **Targeted Interventions**: Knowing which factors most influence student success allows educators to intervene more effectively. For instance, if quiz scores are a strong predictor of success, educators could focus on providing extra quizzes or support for students who struggle in assessments.

6. Innovation:

- **Early Warning System**: The classification model can serve as an early warning system for educators, identifying students at risk of not completing their recommended subjects. By predicting whether a student will complete their courses based on real-time data, institutions can provide timely interventions.
- **Holistic View of Student Success**: The integration of engagement data (such as time spent on the platform and logins) with academic data (quiz performance, past courses) provides a more complete picture of student success, moving beyond traditional academic-only models.