CMPE256 - S23 - Book recommendations

Vinit Kanani
016651323
Kaggle: @iamvinitk

1. Scoring - RMSE
   a. SVD - 1.56784
   b. XGB - 1.64211

2. Data Preprocessing:
   a. Books Data
   The books data had issues with CSV parsing due to escape characters. I used preprocessing techniques to format the data correctly by cleaning up the escape characters and converting the data into a more readable format. This involved removing unnecessary columns and creating new columns that would be more useful in the prediction process.
   b. Users Data
   The user's data had inconsistencies in the format of the location attribute. To address this, I extracted the country from the location attribute and added a new column for the region. Additionally, there were around 40% missing values for the age attribute, so I replaced these values with the average age of the country.
   c. Train Data
   The training data had some ratings above 10, which were replaced with 10 to ensure consistency and fairness in the prediction process.

3. Algorithms
   a. SVD
   SVD (Singular Value Decomposition) is a matrix factorization technique commonly used for collaborative filtering in recommender systems. It decomposes a large user-item rating matrix into three smaller matrices: U, S, and V. The U matrix represents user preferences, the S matrix contains the singular values, and the V matrix represents item attributes.
   I used *GridSearchCV* to find the optimal values for the hyperparameters of the SVD model, which included the number of epochs, the learning rate, and the regularization term.

   b. XGB
   XGB (Extreme Gradient Boosting) is a powerful algorithm used for both classification and regression problems. It is based on decision trees and is known for its speed and accuracy. The *MinMax* Scaler transformed the data so that the values were between 0 and 1, which helped to improve the performance of the model. *Label encoding* was used to transform the categorical features into numerical values, which allowed the XGB model to use them in its calculations. I

used XGB for regression in this project and tuned the hyperparameters using *GridSearchCV*. The hyperparameters included the maximum depth of the tree, the learning rate, and the number of estimators. The best parameters were *learning_rate=01, max_depth=7, n_estimators=1000*.

4. Reference

a. https://www.kaggle.com
b. https://github.com
c. https://scikit-learn.org
d. https://www.tensorflow.org
e. https://towardsdatascience.com/
f. https://xgboost.readthedocs.io/en/latest