**Project Name:** Book Recommendation System

**Github Link:** https://github.com/utkrisht2000/Book-Recommendation-System.git

**Why was this project created?**

Today, a lot of things are recommended to customers based on their interests via recommendation algorithms. One of the most effective strategies for boosting profitability by keeping more consumers in a fiercely competitive market is a recommendation system. Online bookstores like Kindle and Goodreads compete with one another on a variety of fronts. Their technique for recommending books is one of those crucial components. A book suggestion system is made to suggest novels the customer will find interesting.

**What problem is it solving?**

A book recommendation system's goal is to anticipate customers' reading preferences and make appropriate book recommendations. By filtering user reviews, a book recommendation system may take into consideration a variety of factors, including book content and book quality. In some businesses, recommender systems are crucial because, when used well, they may produce considerable revenue or serve as a means of differentiating oneself from rivals. By using this straightforward dataset, associated tasks, and notebooks, we may iteratively explore several recommender algorithm paradigms.

**Entire explanation of project**

* **PROPOSED APPROACH**

The data collection that includes information about the books, the authors, and other pertinent details. Find every null value that appears in our dataset. Additionally, we looked at the data types of each column and verified that our data did not contain any null values. Although the maximum score in our data was 5.0, none of the books in the aforementioned result have a score of 5.0. In fact, we selected these books based on how many notes they had. We took care to ensure that every book in the results shown above had a respectable rating. Our data includes both the average rating and the number of ratings for each individual book. discover the relationship between the quantity of scores and our average score. We are testing this to see whether we can incorporate these columns into our advice. We will also examine how the distribution of average ratings varies with the amount of pages, the language, and the quantity of text reviews in a book.

We will now add a new column with the name "rating between." We'll categorize the ratings in our average rating column into ranges from 0 to 1, 1 to 2, etc. This will function as one of the factors we use into our model to help it provide more accurate forecasts. We will now utilize the Min-Max scaler to lower these values once our features are prepared. This will lessen the prejudice against some of the books that have an excessive number of features. After reducing the number of characteristics, we can now construct our Python-based machine learning book recommendation system utilizing the KNN method. For the purpose of recommending books, we created a machine learning model.

Algorithm for creating next word prediction model :

**Step 1:** Dataset is imported

**Step 2:** Data Exploration

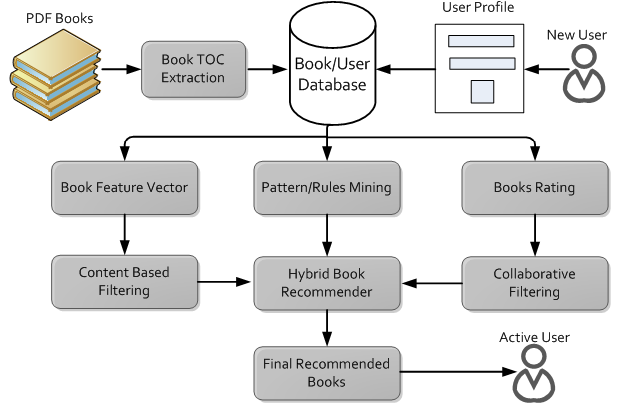
**Step 3:** Data Preparation

**Step 4:** Feature Extraction

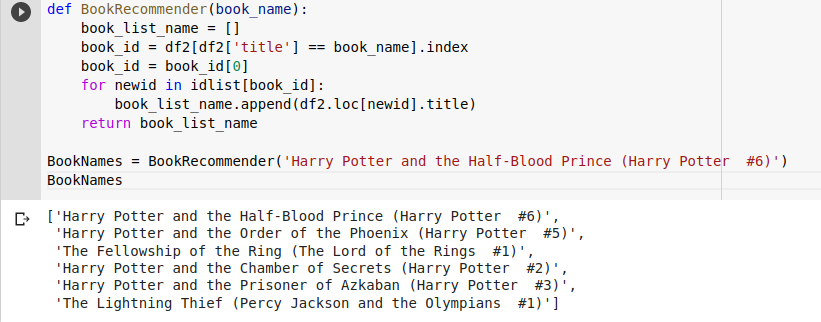
**Step 5:** MinMaxScaling & KNN Implementation

**Step 6:** Testing

* **DATA FLOW DIAGRAM**



* **RESULT**



* **CONCLUSION**

We gave a thorough analysis of the most significant research on deep learning-based recommender systems to yet. We highlighted a number of important research prototypes and presented a classification approach for grouping and arranging current articles. We include some of the most important unresolved issues and exciting potential future developments. In recent decades, there has been a lot of interest in both deep learning and recommender systems as academic areas. Every year, there are a lot of new emerging models and developing strategies. We anticipate that this book will provide readers a thorough knowledge of the essential components of recommender systems.