



# *A Collaborative Filtering Approach to Kindle Book Recommendations*

Name: Punam Kumari  
Roll no.: 30  
Reg no.: 12323153

# Study for recommendation system

- **Content-Based Recommendation:** It is [supervised machine learning](#) used to induce a classifier to discriminate between interesting and uninteresting items for the user.
- **Collaborative Filtering:** [Collaborative Filtering](#) recommends items based on similarity measures between users and/or items. The basic assumption behind the algorithm is that users with similar interests have common preferences.

# Content-Based Recommendation System

Content-based systems recommend items to the customer similar to previously high-rated items by the customer. It uses the features and properties of the item. From these properties, it can calculate the similarity between the items. In a content-based recommendation system, first, we need to create a profile for each item, which represents the properties of those items. The user profiles are inferred for a particular user. We use these user profiles to recommend the items to the users from the catalog.

## Item profile

we need to perform the [TF-IDF vectorizer](#), here TF (term frequency) of a word is the number of times it appears in a document and The IDF (inverse document frequency) of a word is the measure of how significant that term is in the whole [corpus](#).

## User profile

The user profile is a vector that describes the user preference. During the creation of the user's profile, we use a utility matrix that describes the relationship between user and item. From this information, the best estimate we can decide which item the user likes, is some aggregation of the profiles of those items.

## Collaborative Filtering

Collaborative filtering is based on the idea that similar people (based on the data) generally tend to like similar things. It predicts which item a user will like based on the item preferences of other similar users.

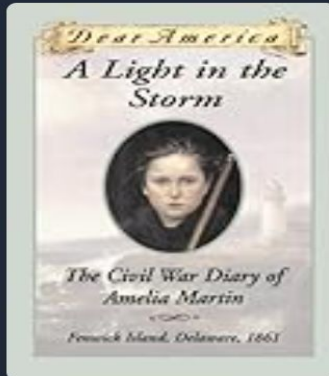
[Collaborative filtering](#) uses a user-item matrix to generate recommendations. This matrix contains the values that indicate a user's preference towards a given item. These values can represent either explicit feedback (direct user ratings) or implicit feedback (indirect user behavior such as listening, purchasing, watching).

- **Explicit Feedback:** The amount of data that is collected from the users when they choose to do so. Many of the times, users choose not to provide data for the user. So, this data is scarce and sometimes costs money. For example, ratings from the user.
- **Implicit Feedback:** In implicit feedback, we track user behavior to predict their preference.

# Top 50 recommended books

My Book

[Home](#) [About](#) [Recommend](#) [Contact](#)



**A Light in the Storm: The Civil War Diary of Amelia Martin, Fenwick Island, Delaware, 1861 (Dear America)**

Author-Karen Hesse  
Votes-4  
Ratings-2.25

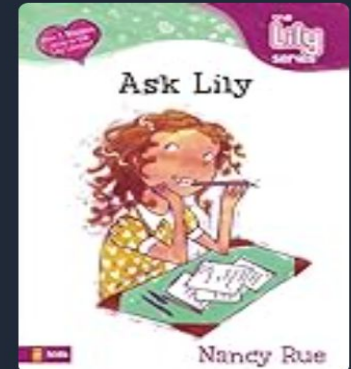


**Always Have Popsicles**

Author-Rebecca Harvin  
Votes-1  
Ratings-0.0

**Apple Magic (The Collector's series)**

Author-Martina Boudreau  
Votes-1  
Ratings-0.0



**Ask Lily (Young Women of Faith: Lily Series, Book 5)**

Author-Nancy N. Rue  
Votes-1  
Ratings-8.0



# Code

## 1. For recommend books

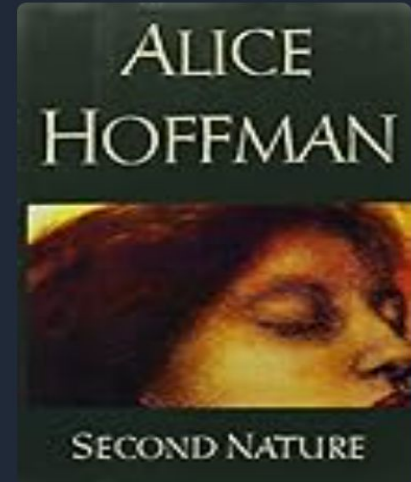
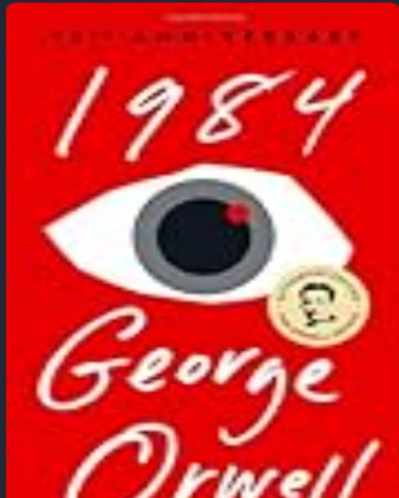
```
def recommend():  
    user_input = request.form.get('user_input')  
    index = np.where(pt.index==user_input)[0][0]  
    distances = similarity_scores[index]  
    similar_items = sorted(list(enumerate(distances)), key=lambda x:x[1], reverse=True)[1:6]  
  
    data=[]  
    for i in similar_items:  
        item = []  
        temp_df = books[books['Book-Title']==pt.index[i[0]]]  
        item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Title'].values))  
        item.extend(list(temp_df.drop_duplicates('Book-Title')['Book-Author'].values))  
        item.extend(list(temp_df.drop_duplicates('Book-Title')['Image-URL-M'].values))  
        data.append(item)  
    print(data)  
    return render_template('recommendation.html', data=data)
```

# Enter book -> recommended book (Animal Farm)

My Recommended Books

[Home](#) [About](#) [Recommend](#) [Contact](#)

Submit



# Link for the project

1. [astrospkc/Pick-book \(github.com\)](https://github.com/astrospkc/Pick-book) -> github link