Abstract:

Now-a-days, everyone depends on reviews by others in many things such as selecting a movie to watch, buying products, reading a book. Recommender systems are used for that purpose only. A recommender system is a kind of filtering system that predicts a user's rating of an item. Recommender systems recommend items to users by filtering through a large database of information using a ranked list of predicted ratings of items. Online Book recommender system is a recommender system for ones who love books. When selecting a book to read, individuals read and rely on the book ratings and reviews that previous users have written. In this paper, Hybrid Recommender system is used in which Collaborative Filtering and ContentBased Filtering techniques are used. The author used Collaborative techniques such as Clustering in which data-points are grouped into clusters. Algorithms such as Kmeans clustering and Gaussian mixture are used for clustering. The better algorithm was selected with the help of silhouette score and used for clustering. Matrix Factorization techniques such as Truncated-SVD which takes sparse matrices as input is used for reducing the features of a dataset. The Content Based Filtering System used a TFIDF vectorizer which took statements as input and returned a matrix of vectors. RMSE (Root Mean Square Error) is used for finding the deviation of an absolute value from an obtained value and that value is used for finding the fundamental accuracy.

Keywords: Book Recommender System, Matrix Factorization, Clustering, K-Means, Gaussian Mixture, Root Mean Square Error..

Introduction:

Introduction Now-a-days, online rating and reviews are playing an important role in books sales. Readers were buying books depending on the reviews and ratings by the others. Recommender system focuses on the reviews and ratings by the others and filters books. In this paper, the Hybrid recommender system is used to boost our recommendations. The technique used by recommender systems is Collaborative filtering. This technique filters information by collecting data from other users. Collaborative filtering systems apply the similarity index-based technique. The ratings of those items by the users who have rated both items determine the similarity of the items. The similarity of users is determined by the similarity of the ratings given by the users to an item. Content-based filtering uses the description of the items and gives recommendations which are similar to the description of the items. With these two filtering systems, books are recommended not only based on the user's behavior but also with the content of the books. So, our recommendation system recommends books to the new users also. In this recommender system, books are recommended based on collaborative filtering technique and similar books are shown using content based filtering. The required dataset for the training and testing of our model is downloaded from Good-Reads website. Matrix Factorization techniques such as Truncated-SVD which takes a sparse matrix of dataset is used for reduction of features. The reduced dataset is used for clustering to build a recommendation system. Clustering is a collaborative filtering technique that is used to build our recommendation system in which data points are grouped into clusters. . In this paper, we used two methods i.e., K-means and Gaussian mixture for clustering the users. The better model is selected based on the silhouette score and used for clustering. Silhouette score or silhouette coefficient is used to calculate how good the clustering is done. Negative value shows that clustering is imperfect whereas positive value shows that clustering was done perfectly. Difference between the mean rating before clustering and after clustering is calculated. Root Mean square Error is used to measure the error between the absolute 2 values and obtained values. That RMSE value is used to find the fundamental accuracy.

Problem Statement

Recommending books using Machine learning algorithms is the main goal of this project. Books are recommended by the clustering model and we are going to train and build using various features such as user's rating, book description, book titles etc. The system groups users into clusters so that each data point within a cluster is similar and dissimilar to the data point in the other cluster. The system we would like to develop will also be able to find an average rating for each cluster and it is going to find top rated books of users from each cluster. All these books shortlisted by our system will be used for training our model in

future. The prediction model needs to be trained so as to produce better results.

The data features are as follows:

Users_dataset.

● User-ID (unique for each user)

● Location (contains city, state and country separated by commas)

● Age

Shape of Dataset - (278858, 3)

Books_dataset.

● ISBN (unique for each book)

● Book-Title

● Book-Author

● Year-Of-Publication

● Publisher

Shape of Dataset - (271360, 8)

Ratings_dataset.

● User-ID

● ISBN

● Image-URL-S

● Image-URL-M

● Image-URL-L

● Book-Rating

Methodology:

System Architecture describes “the overall structure of the system and the ways in which the structure provides conceptual integrity”. The system architecture to build a recommendation system involves the following five major steps.
3.2.1 Data Acquisition
3.2.2 Data Pre-processing
3.2.3 Feature Extraction
3.2.4 Training Methods
3.2.5 Testing Data In Step
3.2.1, Dataset was collected from Good Reads Website in which three datasets are present i.e. Books Dataset, Ratings Dataset, Users Dataset. In Step 3.2.2, Datasets were pre-processed to make them suitable for developing the Recommendation system. In Step 3.2.3, Feature extraction is performed in which Truncated-SVD is used to reduce the features of the dataset and Data splitting is done in which training dataset and testing dataset are divided into 80:20 ratio. In Step 3.2.4, Content Based Filtering System is developed in which book description is taken as an input and Collaborative Filtering System is developed by building a model using K-Means Algorithm over Gaussian Mixture after comparing with Silhouette scores. In step 3.2.5, Testing of models with test data is performed.

• Unsupervised Machine learning algorithms and implementation :

In unsupervised learning, artificial intelligence learns without predefined target values and without rewards. It is mainly used for learning segmentation (clustering). The machine tries to structure and sort the data entered according to certain characteristics. For example, a machine could (very simply) learn that coins of different colors can be sorted according to the characteristic "color" in order to structure them. Unsupervised Machine Learning algorithms are used when the information used to train is neither classified nor labeled. The system does not figure out the right output but it explores the data and can draw inferences from datasets to describe hidden structures from unlabeled data. Unsupervised Learning is the training of machines using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance. Unsupervised Learning is classified into two categories of algorithms:

Clustering:

Clustering is an unsupervised learning method in which we draw references from datasets consisting of input data without labeled responses. Generally, it is used as a process to find meaningful

structure, explanatory underlying processes, generative features, and groupings inherent. Clustering is the task of dividing the population or data points into a number of groups such that data points in the same groups are more similar to other data points in the same group and dissimilar to the data points in other groups. It is basically a collection of objects on the basis of similarity and dissimilarity between them. Clustering is very important as it determines the intrinsic grouping among the unlabeled data present. There are no criteria for good clustering. It depends on the user, what are the criteria they may use which satisfy their need. This algorithm must make some assumptions which constitute the similarity of points and each assumption makes different and equally valid clusters.words. We have represented a dense word-cloud in some of the mostly used words within the corpus.

Clustering Methods:

Density-Based Methods: These methods consider the clusters as the dense region having some similarity and different from the lower dense region of the space. These methods have good accuracy and ability to merge two clusters. Example: DBSCAN (Density-Based Spatial Clustering of Applications with Noise), OPTICS (Ordering Points to Identify Clustering Structure) etc. Hierarchical Based Methods: The clusters formed in this method forms a tree-type structure based on the hierarchy. New clusters are formed using the previously formed one. It is divided into two categories: o Agglomerative (bottom up approach) o Divisive (top down approach) Examples: CURE (Clustering Using Representatives), BIRCH (Balanced Iterative Reducing Clustering and using Hierarchies) etc. Partitioning Methods: These methods partition the objects into k clusters and each partition forms one cluster. This method is used to optimize an objective criterion similarity function such as when the distance is a major parameter 8 example K-means, CLARANS (Clustering Large Applications based upon Randomized Search) etc. Grid-based Methods: In this method the data space is formulated into a finite number of cells that form a grid-like structure. All the clustering operations done on these grids are fast

RESULTS (Screenshots)

Collaborative Filtering (Nearest Neighbor's Based)

```
Books_Name=str(input("Pls Entre the Book Name  \n"))

def recommend_book(book_name):
  book_id=np.where(book_pivot.index ==book_name)[0][0]
  distances,suggestions= model.kneighbors(book_pivot.iloc[book_id, :].values.reshape(1,-1))

  for i in range(len(suggestions)):
    if i==0:
      print("The Suggestions for",book_name,'are :\n\n')
      if not i:
        print(book_pivot.index[suggestions[i]])
recommend_book(Books_Name)
#Example-Books Name - "Harry Potter and the Prisoner of Azkaban (Book 3)"
```

```
Pls Entre the Book Name
Harry Potter and the Prisoner of Azkaban (Book 3)
The Suggestions for Harry Potter and the Prisoner of Azkaban (Book 3) are :


Index(['Harry Potter and the Prisoner of Azkaban (Book 3)',
       'Harry Potter and the Goblet of Fire (Book 4)',
       'Harry Potter and the Chamber of Secrets (Book 2)',
       'Harry Potter and the Sorcerer's Stone (Book 1)',
       'Harry Potter and the Order of the Phoenix (Book 5)'],
      dtype='object', name='title')
```

Collaborative Filtering (Correlation Based)

```
#example= Harry Potter and the Sorcerer's Stone (Harry Potter (Paperback))

isbn = books.loc[books['title'] == bookName].reset_index(drop = True).iloc[0]['ISBN']
row = matrix[isbn]
correlation = pd.DataFrame(matrix.corrwith(row), columns = ['Pearson Corr'])
corr = correlation.join(average_rating['ratingCount'])

res = corr.sort_values('Pearson Corr', ascending=False).head(number+1)[1:].index
corr_books = pd.merge(pd.DataFrame(res, columns = ['ISBN']), books, on='ISBN')
print("\n Recommended Books: \n")
corr_books['title']
```

```
Enter a book name: The Horse and His Boy

 Recommended Books:

0                                  The Last Battle
1       The Voyage of the Dawn Treader (rack) (Narnia)
2                                 The Silver Chair
3    Prince Caspian (rack) : The Return to Narnia (...
4              The Magician's Nephew (rack) (Narnia)
Name: title, dtype: object
```

Collaborative Filtering (User-Item Filtering)

```
k = list(final_rating['title'])
m = list(final_rating['ISBN'])

collaborative = getTopRecommandations(m[k.index(bookName)])
```

```
Input Book:
Harry Potter and the Sorcerer's Stone (Harry Potter (Paperback))

RECOMMENDATIONS:

Harry Potter and the Prisoner of Azkaban (Book 3)
Harry Potter and the Goblet of Fire (Book 4)
Harry Potter and the Order of the Phoenix (Book 5)
Fried Green Tomatoes at the Whistle Stop Cafe
Harry Potter and the Chamber of Secrets (Book 2)
```

Summary:

In EDA, the Top-10 most rated books were essentially novels. Books like The Wild Animus and The Lovely Bones: A Novel .

Majority of the readers were of the age bracket 20-50 and most of them came from North American and European countries namely USA, Canada, UK.

If we look at the ratings distribution, most of the books have high ratings with maximum books being rated 8. Ratings below 5 are few in number.

Author with the most books was Stephen King, Nora Roberts and James Patterson.

A recommendation system helps an organization to create loyal customers.

Future Scope:

Given more information regarding the books dataset, namely features like Genre, Description etc., we could implement a content-filtering based recommendation system and compare the results with the existing collaborative-filtering based system.