Implementing Matrix Factorization Technique For Collaborative Filtering Based Recommendation System For Online Book Suggestions

First Author: Nupur Singh (Email: nupursingh.scsebtech@galgotiasuniversity.edu.in)
Second Author: Yash Chauhan (Email: yashchauhan.scsebtech@galgotiasuniversity.edu.in)
Author: Rahul Anjana (Email: rahul.anjana@galgotiasuniversity.edu.in)

I. ABSTRACT

A recommender system is a form of filtering device that predicts a user's score of an item. Recommender structures suggest gadgets to users via filtering the rough a large database of information via the usage of a ranked listing of anticipated rankings of objects. When choosing a book to look at, humans have a look at and rely upon the e-book scores and critiques that previous customers have written. In this paper, a hybrid recommender device is used wherein collaborative filtering and content-based total filtering strategies are used. The datasets are used, which may be downloaded from the Good reads net website, which includes the functions of customer s and e-books. The content primarily based filtering device makes use of the simplest functions for recommendation, while the collaborative filtering gadget uses e-books and patron features, and the consequences are displayed on the front end.

Keywords: - Book Recommender System; Truncated-SVD; Clustering; Root Mean Square Error.

II. INTRODUCTION

A hybrid recommendation system is used to decorate our guidelines. The technique utilized with the aid of advice systems is collaborative filtering. This technique filters information with the aid of collecting information from other customers. Collaborative filtering systems observe the similarity index based technique. The ratings of those gadgets by the users who have rated every item decide the similarity of the gadgets. The similarity of customers is determined by the similarity of the rankings given by the customers to an object. Content-based total filtering uses the definition of the devices and offers tips that are similar to the gadgets themselves. With these filtering systems, books are endorsed not only based on the behaviour of customers, but additionally on the content of the books. As a result, our advice tool additionally suggests books to new clients. In this paper, we used two techniques, i.e., adequate approach and Gaussian aggregate, for clustering the customers. The root-implied square errors are used to measure the distinction between the absolute values and purchased values. The RMSE price is used to determine the critical accuracy.

III. LITERATURE REVIEW

To recommend books, most researchers used Pearson's Correlation Coefficient function to calculate similarity among book rankings. Avi Rana and Ok. Deeba, et. al. (2019) [1] proposed a paper "on-line book advice device: the use of Collaborative Filtering (With Jaccard Similarity)". In this paper, the author used CF with Jaccard similarity to get greater correct tips because widespread CF difficulties are scalability, sparsity, and cold starts. So, to conquer these difficulties, they used CF with Jaccard Similarity. JS is based totally on a pair of books index that is a ratio of commonplace users who've rated both books divided by the sum of customers who have rated books for my part. Books with an excessive JS index are surprisingly encouraged.

G. Naveen Kishore, et al. (2019) [2] proposed a paper "online e-book advice system". The dataset used in this paper was taken from the website "desirable books-10k dataset", which includes ten thousand precise books. The book_id, user_id, and rating are the features. In this paper, the writer followed a Keras deep studying framework model to create neural network embedding. Uko E Okon, et. al. (2018) [3] proposed a paper "An advanced online book recommender gadget using a collaborative filtering algorithm". The authors designed and evolved a recommendation model by using a short set of rules, collaborative filtering, and an item-oriented evaluation and design technique (OOADM). This device produces an accuracy of 90-ninety five%. Jinny Cho, et al. (2016) [4] proposed a paper "e-book advice system". In this paper, the writer uses two approach methods, which are content-based (CB) and collaborative filtering (CF). They used two algorithms, UVDecomposition and k-Nearest acquaintances (KNN). They obtained an end result with an accuracy of 85%. Sushma Rjpurkar et al. (2015) [5] proposed a paper "e book recommendation gadget". In this paper, the writer used Associative Rule Mining to find association and correlation relationships amongst a dataset of objects. They used CB and CF tactics to construct a system. Abhay E. Patil, et al. (2019) [6] proposed a paper "on-line e-book recommendation machine using association rule mining and collaborative filtering." The author detects frequent going on styles and correlations and shapes associations using various databases, including relational and transactional databases. They used two approaches, i.e., user-primarily based and item-based collaborative filtering, and used the Pearson correlation coefficient to locate similarity between the items.

IV. THE PROPOSED SYSTEM

This paper is split into five sections. In segment 3.1, datasets turn out to be amassed from the specific Reads website, wherein three datasets are given, i.e., the Books Dataset, Rankings Dataset, and Clients Dataset. In section three.2, datasets were preprocessed to make them suitable for developing the advice gadget. Truncated-SVD is used to reduce the talents of the dataset. Phase three.three is entirely content-driven. An ebook filtering device has advanced in which an e-book description is taken as an input. In segment three.4, record splitting is accomplished, wherein the schooling and check datasets are divided into 80:20 racial.

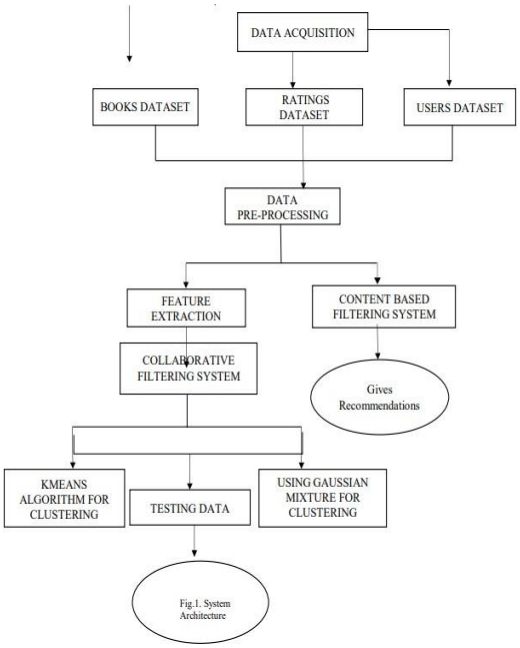

Fig.1. System Architecture

V. DATABASE

V.1. Dataset

The first step is to find the dataset for building our gadget. We need to identify the dataset that gives green results for our gadget. In this step, we used to acquire distinct datasets that might be to be had in record repositories, and the records are probably uncooked facts that are saved in the shape of zip files and databases. The most important aspects of discovering the truth are pleasant and quantity. The outcomes are more correct if we have more factors within the database. We accrued the 3 datasets, which include capabilities required for the machine. The three datasets are the Books dataset, Customer dataset, and Rankings dataset. The Books dataset contains functions, particularly book_authors, which gives the call of the author; book_desc, which offers an outline of books; book_format, which tells about the packing of books; book_rating, which suggests a median rating of the book; book_rating_count, which gives a complete variety of rating matters; book_title, which suggests the title of the e book; genres, which suggests genres of e-book; image_url, which includes the URL of the e-book cowl web page and the score The dataset contains features, namely person-id, book-id, and score

V.2. Pre-Processing data

At this step, the primary aim is to understand the features of the information that the writer has to work with. In this paper, we implemented Truncated-SVD, which is available for sparse matrices to lessen the dimensionality. By way of the use of the Truncated-SVD, the capabilities of the dataset are decreased. The initially accrued data includes noise, missing values, and copy values, which cannot be directly used for machine learning algorithms. Records preprocessing is the method of cleansing the records, casting off duplicates, and filling the null values with the average of that attribute. In that manner, the final dataset turned into the preparation for building the version that gives an efficient performance. We cut up the information into 80:20, i.e., 80% of the dataset is used to train the advanced version and 20% of the facts are used to check the model.

V.3. Content Based filtering System

A content-based filtering device recommends items which might be much like the content of the object. This device uses the description of the objects and offers hints that are much like the outline. We used cosine similarity as a similarity characteristic for this gadget. The object-content matrix, which describes the attributes of the capabilities, is taken as an input. Based on the perspective among the vectors, cosine similarity is calculated. We improve the user experience of the content primarily based device by normalising and tuning the attributes with the TF-IDF vectorizer. TF (time period frequency) is termed as a word frequency in a record. IDF (Inverse Report Frequency) is the universe file frequency

V.4. Training data

In Data processing, we split the data into training data and testing data so that we can train and test the model. By this method, we can get more performance of our machine learning model. If the total dataset is used to train the model and then we have to test the model with a completely different dataset. So, we would get more difficulties to understand the correlation between the models with different sets which decreases the performance. That's why we have to use the same data for training and testing to get better performance. Training is very essential to understand various features and patterns.

V.5. Collaborative Filtering System

The Collaborative Filtering gadget collected and analysed a large quantity of person-primarily based records and predicts the person's choice primarily based on their similarity with other users. The person-based collaborative filtering method finds comparable users to the modern-day consumer based entirely on ratings for given objects and then predicts the score for an object by calculating the rankings given by the comparable customers for that item. model-based collaborative filtering uses dimension reduction by building a user-object matrix with users as rows, gadgets as columns, and the scores of customers for an object as values. In this paper, we used Truncated-SVD for dimensionality discount.

V.6. K-Means Clustering

K-Means Clustering is an Unsupervised Learning algorithm, which groups the unlabelled dataset into different clusters in such a manner that each dataset belongs to only one group that has similar features. Here K defines the number of pre-defined clusters. We have to associate each cluster with a centroid in this algorithm. The sum of distances between the data point and their corresponding clusters should be minimized. The unlabelled dataset is taken as an input and the dataset into k-number of clusters is divided, and the process is repeated until it does not find the best clusters. We have to predetermine the k value in this algorithm. Elbow method is used to find the value of k which decides the number of clusters. This method uses the Within Cluster Sum of Squares (WCSS) value that defines the total variations within a cluster. The Formula for calculating the value of WCSS for n clusters is given by **eq-1**.

$$WCSS = \sum Pi\ in\ Cluster1 distance\ (Pi\ C1)2 + \sum Pi\ in\ Cluster2 distance (Pi\ C2)2 + \cdots + \sum Pi\ in\ Cluster\ n\ distance (PiCn)2$$

(eq -1)

Algorithm steps:

Step-1: Select the number K which gives the number of clusters.

Step-2: Select random K number of points or centroids.

Step-3: Each data point to their nearest centroid should be assigned, which forms the predefined K clusters.

Step-4: Calculate the variance and place a new centroid for each cluster.

Step-5: We have to repeat the step-3, each data-point to the new closest centroid of each cluster should be reassigned.

Step-6: If reassignment happens, then go to step-4 or else go to step-7.

Step-7: Stop.

V.7. Gaussian Mixture

Gaussian mixture fashions are effective clustering algorithms. It assumes that there are a certain number of Gaussian distributions where each distribution represents a cluster. This version groups the information factors collectively into a single distribution. These models used the gentle clustering technique for assigning statistical factors to Gaussian distributions. In a one-dimensional space, the possibility density feature of a Gaussian distribution (univariate) is given by using **eq-2.**

$$P(x \mid \mu, \sigma 2) = N\ (x;\ \mu, \sigma 2) = 1/\ Z\ [\exp(-\ (x\ -\ \mu)\ 2/\ 2\sigma 2\ )]$$

-eq2

Where Z is the normalization constant i.e. , $Z = \sqrt{2\pi\sigma 2}$, μ is the mean i.e., $\mu = E[x]$, and σ2 is the variance of the distribution i.e., $\sigma 2 = var[x]$.

In a multidimensional space, the probability density function of a Gaussian distribution (multivariate) is given by **eq-3**.

$$P(x \mid \mu, Z) = N\ (x;\ \mu, Z) = 1/Z\ [exp(-\ 1/2\ (x - \mu)TZ - 1(x - \mu))]$$

.....**eq-3**

Where X is the input vector, μ is the 2D mean vector, and Σ is the 2×2 covariance matrix. Thus, we would have K (number of clusters) Gaussian distributions.

VI. PERFORMANCE ANALYSIS AND RESULTS

In this step, we analyse the performance of a model and checks the results. While clustering, we have applied K Means and Gaussian mixture to cluster the users. The model which gave the best silhouette score would be used for constructing the device. A Silhouette score is used to test the efficiency of clustering. The tremendous price of the Silhouette rating suggests top clustering while poor value i ndicates terrible clustering. First, we built the model by using the okay-manner method, after which we fitted and are expecting the model with the education dataset. Then, we calculate the silhouette score with the advanced version.

```
Clusterer_KMeans=KMeans(n_clusters=7)
.fit(book_ratings_training)

Preds_KMeans=Clusterer_KMeans.predict (book_ratings_training)

Kmeans_score=silhouette_score(book_rati
ngs_training,preds_KMeans)
```

Print(Kmeans_score)

Now, we built the model by using Gaussian mixture and then we fit and predict the model using same training dataset. Then, we calculate the silhouette score with the developed model.

	K-Means	Gaussian Mixture
Silhouette Score	0.043396833258441 1	0.01677887231368893 3

Table-1: Silhouette scores of two model

After checking the two ratings, we don't forget the version the usage of the okay means approach with cluster be counted at seven due to the higher score. Then, this version is used to cluster the customers.

The difference between the average implied rankings of books in step with the check consumer and the average mean score of books for the cluster's favoured is calculated. The average suggest score is computed as follows: average suggest.

$$score = Sum(scores)/len(scores)$$

Mean rating for 10 random books	3.8876949740034736
Mean rating for 10 books of cluster's favourites	4.3735008665511135

Table-2: Comparing Mean ratings

VII. CONCLUSION

In this paper, we advocated the books for a consumer using the ok-manner version clustering, which is an unsupervised system learning algorithm. We compared the 2 methods for clustering and recognised the satisfactory version through calculating the silhouette score. We used the datasets that are downloaded from the Good Reads website. We implemented the functionalities within the system according to the necessities after understanding all of the modules of the device.

VIII. REFERENCES

*1+ Avi Rana and K. Deeba et.al, "Online Book Recommendation System using Collaborative Filtering (With Jaccard Similarity)" in IOP ebooks 1362, 2019.

[2] G. Naveen Kishore, V. Dhiraj, Sk Hasane Ahammad, Sivaramireddy Gudise, Balaji Kummaraa and Likhita Ravuru Akkala, "Online Book Recommendation System" International Journal of Scientific & Technology Research vol.8, issue 12, Dec 2019.

[3] Uko E Okon, B O Eke and P OAsaga, "An Improved Online Book Recommender System using Collaborative Filtering Algorithm", International Journal of Computer Application vol.179-Number 46, 2018.

[4] Ms. Sushma Rjpurkar, Ms. Darshana Bhatt and Ms. Pooja Malhotra, "Book Recommendation System" International Journal for Innovative Research in Science & Technology vol.1, issue 11, April 2015.

[5] Abhay E. Patil, Simran Patil, Karanjit Singh, Parth Saraiya and Aayusha Sheregar, "Online Book Recommendation System using Association Rule Mining and Collaborative Filtering" International Journal of Computer Science and Mobile Computing vol.8, April 2019.

[6] Suhas Patil and Dr. Varsha Nandeo, "A Proposed Hybrid Book Recommender System" International Journal of Computer Applications vol.6 – No.6,Nov– Dec 2016.

[7] Ankit Khera, "Online Recommendation System" SJSU ScholarWorks,Masters Theoremand Graduate Research,Master's Projects, 2008.

[8] Anagha Vaidya and Dr. Subhash Shinde, "Hybrid Book Recommendation System" International Research Journal of Engineering and Technology (IRJET), July 2019.

[9] Dhirman Sarma, Tanni Mittra and Mohammad Shahadat Hossain, "Personalized Book Recommendation System using Machine Learning Algorithm" The Science and Information Organization vol.12,2019.