Avalokiteshvara Journal of Artificial Intelligence

http://hcapit.org/ajai.html

ISSN: XXXX-XXXX (Pending)

Research Article

Comparative Study of Item-Based Collaborative Filtering Algorithms for Book Recommendation Systems

Shradha U. Lipane[1]

[1]CSMSS Polytechnic, Chhatrapati Sambhajinagar, MS, India.

lipane05@gmail.com

Corresponding Author Email: lipane05@gmail.com

Abstract

Book recommendation systems pivotal in enhancing user experience and engagement across digital platforms, facilitating personalized content discovery in a vast sea of information and choices. This paper presents a comprehensive study on the implementation and evaluation of item-based collaborative filtering algorithms for book recommendation systems. Two prominent algorithms, k-nearest neighbor (KNN) and singular value decomposition (SVD) are used with cosine similarity as a similarity metric. This study compares these algorithms in terms of their effectiveness in recommending books that are relevant to users. The KNN algorithm identifies the nearest neighbors of a given book based on user ratings, while SVD decomposes the user-item interaction matrix to capture latent features underlying the data. Both algorithms offer unique advantages and trade-offs, which are thoroughly analyzed in this study. The evaluation metrics include precision providing insights into the accuracy and effectiveness of the recommendation models. Additionally, the computational efficiency of each algorithm is assessed to understand its scalability in real-world applications.

Keywords: Item-based collaborative filtering algorithm, k-nearest neighbor, singular value decomposition, cosine similarity.

1. Introduction

Recommendation systems are algorithms and techniques used in information filtering to suggest items or information to users based on their preferences, behaviors, and interests. These systems are widely used in various online platforms such as movies, books, music, products, etc., suggest relevant items to user. The recommendation systems have found applications beyond commerce and entertainment, entering areas such as education.

Collaborative filtering is the most useful approach to the recommendation system that depends on analyzing user interactions and preferences to identify patterns and make recommendations based on similar user's behavior. Item-based collaborative filtering technique focuses on the similarity between items to generate recommendations used for improving the accuracy and performance of the model. In this framework, the similarity between pairs of items is quantified using metrics such as cosine similarity used to construct a user-item similarity matrix. The item-based collaborative filtering involves machine learning algorithms k-nearest neighbors and singular value decomposition.

KNN is used with the item-based collaborative filtering to identify the k most similar items to a given item. similarity between items is calculated using distance metrics cosine similarity. Once the k nearest neighbors are identified, their ratings are used to generate recommendations for the user. Final recommendation based on an average of the ratings of the nearest neighbors.

To improve the accuracy the item-based collaborative filtering algorithm by combining item-based algorithm with Singular Value Decomposition (SVD) to decrease the dimension number of the user-item rating matrix. SVD is applied to decompose the user-item interaction matrix into three lower-dimensional matrices: user matrix, item matrix, and a diagonal matrix of singular values.

In item-based collaborative filtering with KNN and SVD algorithms are effective approaches to building recommendation systems which help users to find the relevant item from the dataset with less effort and less spending time with more accurate items. In this paper we proposed an item-based book recommendation system by using KNN and SVD algorithms, our contribution includes KNN is used to measure item-item similarity which enhances the performance of the recommendation system. Additionally, SVD is used to reduce dimensionality which improves the quality of recommendations, this algorithm demonstrates the effectiveness, accuracy and quality of recommendation system.

2. Literature Review

The importance of item-based collaborative filtering is it gives personalized recommendations based on similarities between items rather than users and this approach also handle large datasets which makes it more practical for real-world applications. Item-based collaborative filtering uses rating data between users to get recommendations. Collaborative filtering is a widely used recommendation algorithm, focusing on user interests and behavior analysis for book recommendations various collaborative filtering algorithms, such as user-based and item-based approaches, have been explored in the literature to enhance recommendation accuracy and system performance. Studies have highlighted the importance of user feedback and similarity calculations in improving recommendation systems, particularly in the context of book selection and library applications [1].

Collaborative filtering algorithms, such as user-based and item-based approaches, are extensively used for book recommendations. User feedback plays a crucial role in enhancing recommendation accuracy and system performance [2]. Collaborative filtering algorithms, particularly item-based approaches, are widely utilized for personalized recommendations in online platforms. Studies emphasize the importance of similarity calculations and user behavior analysis in improving recommendation systems, especially in e-commerce and marketplace settings [3]. In studies emphasize the significance of similarity calculations and user behavior analysis in improving recommendation systems, particularly in library applications. Research explores the implementation of item-based collaborative filtering algorithms to enhance book recommendations based on user feedback and ratings [2][4]. Item-based collaborative filtering algorithms are utilized for recommendations. User feedback and ratings are essential for enhancing recommendation accuracy and system performance in recommendation systems. Research highlights the significance of user preferences and community opinions in developing effective recommendation systems for various products and services [5].

According to [6], research explores the implementation of item-based collaborative filtering algorithms to enhance book recommendations based on user preferences and community opinions. The collaborative filtering method employed in the system facilitates efficient recommendation generation by leveraging user feedback and community opinions to enhance the quality of book suggestions The system's design, which encourages users to rate books, add them to reading lists, and provide feedback, offers opportunities for continued user engagement and improvement of recommendation accuracy.
As discussed in [7], the utilization of collaborative filtering algorithms, focusing on item-based approaches for book recommendations. The study highlights how user ratings play a crucial role in enhancing recommendation accuracy and system performance. Efficient recommendation systems require a thorough analysis of user behavior and similarity calculations to improve recommendation quality, especially in book recommendation contexts. Recommendation Algorithm based on MARM, which involves simplifying the User-Attribute Item Attribute Rating Matrix to derive MARM for recommending new items or users. The methodology involves comparing the proposed algorithms with traditional collaborative filtering and other recommendation algorithms to assess their effectiveness in enhancing recommendation accuracy.

As stated in [8], the challenges faced by recommendation systems, such as the cold start problem and matrix sparsity, and solutions to enhance system stability and accuracy. They aim to prove the effectiveness of incorporating TensorFlow techniques in recommendation systems to address these challenges and improve user experience by recommending relevant books efficiently. The importance of TensorFlow techniques in improving recommendation accuracy and user satisfaction in book recommendation systems. It discusses the efficiency of online recommendation systems in quickly providing relevant information to users based on their preferences [8][9].

In prior studies, the importance of personalized recommendation systems in higher education where they offer a comparative analysis of different collaborative filtering techniques, showcasing the effectiveness of matrix factorization collaborative filtering, particularly the SVD algorithm, in outperforming neighborhood-based recommendation like the KNN model. The significance of hyperparameter tuning and dataset pruning in enhancing the performance of the recommendation system, showcasing the iterative process of model optimization to achieve optimal accuracy scores [10]. The purpose of conducting a literature review on recommendation systems for building a book recommendation system using machine learning algorithms is to gain a comprehensive understanding of the existing methodologies, techniques, and challenges in the field. By reviewing prior research, one can identify the most effective algorithms, approaches, and evaluation metrics for developing an accurate and efficient book recommendation system.

3. Methodology

A recommendation system is an automatic suggestion system that provides suggestions on books just like our friends, relatives, and neighbors give suggestions. It helps user to find interesting items, reduce number of options items, discover new items which relevant to the user. Recommendation systems also help businesses to increase sales, increase user satisfaction, understand user needs. Here Figure 1. represents our proposed recommendation model architecture.

1) Data collection
During this phase wide range of data is collected for the recommendation system including here Goodreads Dataset which provides us to access information about books, users, and ratings. book information are includes details about the product (title, isbn, author, publication year, publisher etc., user information it refers to personal information (age and location) of the user and user interactions are the feedback data that includes user ratings provided by users this help in understanding the user interest.

2) Data Pre-processing
Here Book Crossing Dataset contains duplicates, and missing values. To build a recommendation model data preprocessing is an important task, first the cleaning process is done by handling missing values and dropping out rows that contain duplicate values from the data, after cleaning dataset we exclude books that have a 0 rating because it o rating means the user dislikes that book and if this book get recommended to the user and this uncertainty can affect the accuracy of the recommendation model. To avoid this issue we take books that are rated by at least 10 users and users that have rated at least 25 different books. The data preprocessing makes the dataset suitable for a system which also helps to increase the accuracy and efficiency of the model.

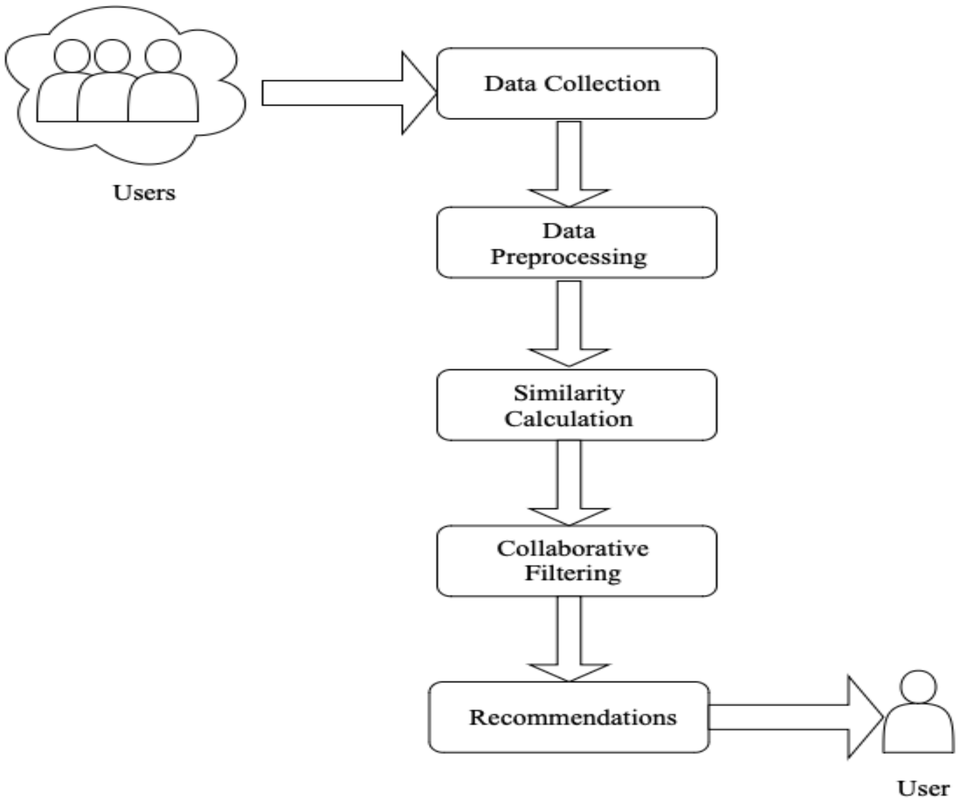

Figure 1. Proposed Recommendation System Architecture

3) Similarity Calculation

The data prepared to effectively analyze user-item ratings, we created a pivot table where rows represent the book titles, columns represent user id and values are ratings given by user to the item. This table shows which item is rated by which user and shows the corresponding rating of that user given to books. By preparing this user-item rating data we compute the similarity between items using similarity metrics i.e. cosine similarity. Cosine similarity measures the similarity between items based on their ratings.

4) Collaborating Filtering

Collaborative filtering is a technique used to generate recommendations based on user-item-rating. Two approaches used by collaborative filtering are k nearest neighbors and single-valued decomposition. KNN algorithm is used to identify nearest neighbors for each user or item based on user rating. For each item in the dataset, identify its k most similar items based on the item similarity matrix. In comparison SVD algorithm used in the recommendation system, SVD is used to uncover latent factors that represent underlying patterns in user-item interactions. By decomposing the user-item interaction matrix into lower-dimensional representations, SVD effectively captures similarities and relationships between users and items. It predicts missing ratings by reconstructing the original matrix using the reduced representations. In collaborative filtering, SVD identifies similar users or items based on these representations, facilitating accurate recommendation generation.

5) Recommendations

For a target user, KNN predicts the rating or preference for items by aggregating the ratings of the nearest neighbors. These predicted ratings are then used to recommend the top n items with the highest predicted ratings to the user. SVD predicts missing ratings for user-item pairs by reconstructing the original matrix using the reduced representations. Based on these predicted ratings, SVD recommends the top-N items with the highest predicted ratings to the user. Both KNN and SVD algorithms produce predicted ratings or preferences for user-item pairs, which are then utilized to generate personalized recommendations for users in a recommendation system.

4. Item Based Collaborative Filtering

The collaborative filtering recommendation system is recommending new items or predicts the rating of specific target item for users according to their preferences. Finding similar items of target items is a crucial task in the item-based recommendation system. There are several ways to calculate similarity between items by using commonly used methods and cosine similarity [8].

1) Cosine Similarity

Cosine similarity is a metric used to measure similarity between two items based on their ratings. By collecting user ratings for books data which consists of a matrix where rows represent books and columns represent user, with the cells containing the ratings given by users to books. Each cell (*i,j*) in the matrix can contain the rating that user *i* has given to item *j*. Cosine similarity measures the cosine of the angle between two non-zero vectors in an n-dimensional space.

$$\text{Cosine similarity}(i, j) = \frac{\sum_u r_{u,i} \cdot r_{u,j}}{\sqrt{\sum_u r_{u,i}^2} \cdot \sqrt{\sum_u r_{u,j}^2}} \qquad (1)$$

where -
ru,i is the rating of user *u* for item *i* in the user-item matrix,
ru,j is the rating of user u for item *j* in the user-item matrix,
the user-item matrix as input and returns a matrix where each element (*i,j*) represents the cosine similarity between item *i* and item *j*.

2) KNN-based collaborative filtering

The KNN model was initialized after computing the similarity matrix. This involves specifying the number of nearest neighbors *k* to consider when finding similar items. To identify nearest neighbors, we identified the k nearest neighbors for each item *i* based on calculated similarity score. After identifying the *k* value, we denote the set of *k* which most similar items to items. This step involves storing the similarity information within the model, enabling fast retrieval of nearest neighbors during the prediction phase. To predict the user on the item we used a weighted average of the rating of KNN as follows,

$$\hat{r}_{ui} = \frac{\sum_{j \in N_i^k(u)} \text{similarity}(i,j) \cdot r_{uj}}{\sum_{j \in N_i^k(u)} |similarity(i,j)|} \qquad (2)$$

where -
$\sum_j \in N_i^k\,(u)$ is the set of *k* items that are most similar to item *i* that user *u* has rated.
$\hat{r}_{ui}$ is ratings to predict.

When we want to find similar items to a given item then the recommendation model is fitted with KNN and The model retrieves the k items with the highest similarity scores from the similarity matrix. The retrieved nearest neighbors can then be used for generating recommendations for users based on similar items. The KNN algorithm is used to efficiently find the most similar items to a given item by querying the computed similarity matrix and generating recommendations.

3) SVD-based collaborative filtering

When it comes to dimensionality reduction, in machine learning the Singular Value Decomposition (SVD) is a popular system of matrix factorization method. Such a method shrinks the space dimension from N-dimension to K-dimension (where K<N) and reduces the number of features. SVD constructs a user-item matrix where the row contains users and columns contain items and the elements are given by the users' ratings. Singular value decomposition decomposes a matrix into three other matrices and extracts the factors from the factorization of a high-level (user-item-rating) matrix. Singular value decomposition is the popular matrix decomposition. SVD decomposes this user-item interaction matrix into three matrices: U, Σ, and V^T.

U (m x r): The left singular vectors matrix represents how users are related to latent features.
Σ (r x r): The diagonal matrix of singular values represents the strength of each latent feature.
V^T (r x n): The right singular vectors matrix represents how items are related to latent features.

The decomposition can be represented as,

$$A \approx U\Sigma V^T \quad (3)$$

where A is the original matrix,
U and V are orthonormal matrix and
Σ is a diagonal matrix containing the singular values.

5. Experiments and Results

This chapter presents the experimental analysis conducted to evaluate the performance of our proposed recommendation system using KNN and SVD algorithms. Book crossing dataset which was accessed from Kaggle https://www.kaggle.com/datasets/syedjaferk/book-crossing-dataset (accessed May 17, 2023) comprises three CSV files books, users and ratings which contains 242134 books and 1048575 ratings given by 95513 user where rating range is 1-10. To evaluate the performance of the system effectively data is split into training and testing sets.

The dataset is randomly split into 80% training data and 20% testing data, By using KNN and SVD we build a book recommendation model. Both models are tested with unseen datasets that are taken from Kaggle, the evaluation of this testing is measured by accuracy and precision measures. The descriptive statistics of the dataset are shown in the above Table I. This statistical summary reveals the standard deviation, mean rating, median rating, and rating range. It also describes the strong tendency towards high ratings. A large standard deviation and a wide range of users suggest a varied and extensive user base.

After building a recommendation model using SVD and KNN using known data, both models were evaluated using unknown datasets. These experiments revealed a clear advantage of the SVD model over KNN when tested on unknown datasets. SVD and KNN both models achieved better performance on unknown datasets. Comparative analysis is conducted to understand the differences in system performance between known and unknown data. To compare the efficiency of two models KNN and SVD, using four unknown datasets. All datasets are sourced from Kaggle. Both models were evaluated using accuracy precision. Here we have represented the result of performance metrics scores of different datasets in given figures respectively. It can see that both models demonstrate a high level of effectiveness in generating books. The accuracy plot shows the performance of testing of both KNN and SVD using unseen datasets, Figure 2. shows how both model's overall accuracy for this data shows effectiveness of both models. Here we represent the precision of unseen datasets, As shown in Figure 3. The precision score of all four dataset shows how both models deliver more effective recommendations to the users.

	user_id	book_rating
count	55211.000000	55211.00000
mean	123624.66225	7.922262
std	74130.760296	1.746134
min	254.000000	1.000000
25%	59971.000000	7.000000
50%	122793.00000	8.000000
75%	187256.00000	9.000000
max	278633.00000	10.000000

Table 1. Statistic summary of data

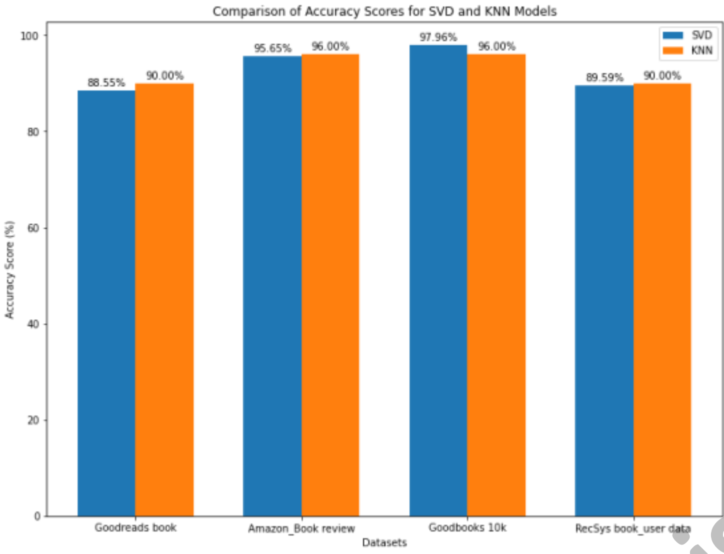

Figure 2. Accuracy for unseen data

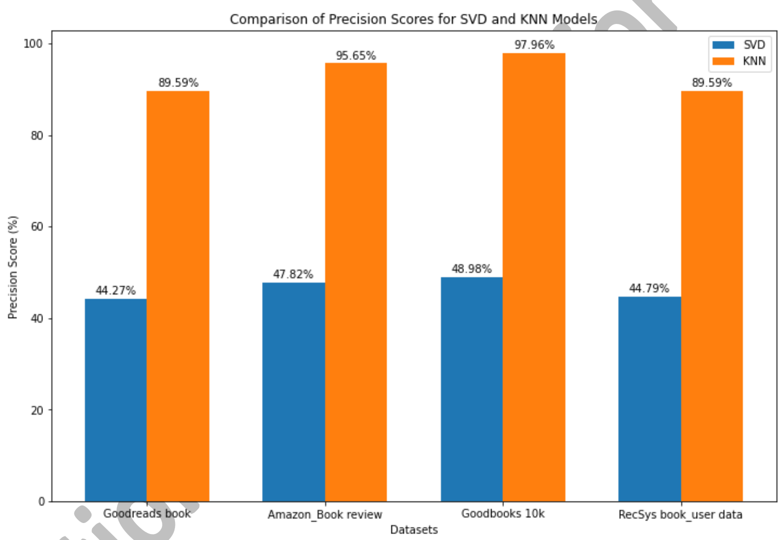

Figure 3. Precision for unseen data

Table 2: Results of unseen data

Algorithms	**SVD**		**KNN**	
Datasets	**Accuracy**	**Precision**	**Accuracy**	**Precision**
Goodreads book	88.56%	42.27%	90.0%	89.59%
Amazon book review	95.65%	47.82%	96.0%	95.65%
Goodbooks 10k	97.96%	48.98%	96.0%	97.96%
RecSys book user data	89.59%	44.79%	90.0%	89.59%

With unseen datasets both KNN and SVD work effectively to retrieve relevant items. With high accuracy, both models effectively enhance user satisfaction by providing relevant books. The performances of SVD and KNN recommendation models are compared. So based on this SVD and KNN-based item-based recommendations give better results on unseen data. This research gives an effective recommendation model with accuracies between 87% to 97% on unseen data. The results are shown in Table 2.

5. Conclusions

Our recommendation system is tested on the bookcrossing dataset which contains users, books and ratings data. In this paper, a book recommendation system is proposed using item-based collaborative filtering method for top n recommendation. Item-based collaborative filtering-based recommendation model is created using popular machine learning algorithms i.e. SVD and KNN. Both models are tested using four datasets to evaluate the performance of the model. In comparison of both model KNN give better performance in terms of accuracy and precision. This indicates that both models generate relevant and accurate recommendations.

References

[1] N. K. Kusumawardhani, M. Nasrun and C. Setianingsih, "Web Recommended System Library Book Selection Using Item Based Collaborative Filtering Method," *2019 IEEE International Conference on Engineering, Technology and Education (TALE)*, Yogyakarta, Indonesia, 2019, pp. 1-8, doi: 10.1109/TALE48000.2019.9225960.

[2] B. Ujkani, D. Minkovska and L. Stoyanova, "A recommender system for WordPress themes using item-based collaborative filtering technique," *2020 XXIX International Scientific Conference Electronics (ET)*, Sozopol, Bulgaria, 2020, pp. 1-3, doi: 10.1109/ET50336.2020.9238305.

[3] P. Venil, G. Vinodhini and R. Suban, "Performance Evaluation of Ensemble based Collaborative Filtering Recommender System," *2019 IEEE International Conference on System, Computation, Automation and Networking (ICSCAN)*, Pondicherry, India, 2019, pp. 1-5, doi: 10.1109/ICSCAN.2019.8878777.

[4] G. Prabowol, M. Nasrun and R. A. Nugrahaeni, "Recommendations for Car Selection System Using Item-Based Collaborative Filtering (CF)," *2019 IEEE International Conference on Signals and Systems (ICSigSys)*, Bandung, Indonesia, 2019, pp. 116-119, doi: 10.1109/ICSIGSYS.2019.8811083.

[5] A. K. Sahoo, C. Pradhan and B. S. Prasad Mishra, "SVD based Privacy Preserving Recommendation Model using Optimized Hybrid Item-based Collaborative Filtering," *2019 International Conference on Communication and Signal Processing (ICCSP)*, Chennai, India, 2019, pp. 0294-0298, doi: 10.1109/ICCSP.2019.8697950.

[6] Wu, C. M., Garg, D., & Bhandary, U. (2018). "Movie Recommendation System Using Collaborative Filtering". 978-1-5386-6565-7118/$31.00 ©2018 IEEE. https://doi.org/10.1109/icsess.2018.8663822.

[7] N. Ifada, M. K. Sophan, M. N. Fitriantama and S. Wahyuni, "Collaborative Filtering Item Recommendation Methods based on Matrix Factorization and Clustering Approaches," *2020 10th Electrical Power, Electronics, Communications, Controls and Informatics Seminar (EECCIS)*, Malang, Indonesia, 2020, pp. 226-230, doi: 10.1109/EECCIS49483.2020.9263450.

[8] F. Marisa et al., "Performance comparison of collaborative-filtering approach with implicit and explicit data," International Journal of Advanced Computer Science and Applications, vol. 10, no. 10, 2019. doi:10.14569/ijacsa.2019.0101016.

[9] Al-Ghamdi, M., Elazhary, H., & Mojahed, A. (2021). "Evaluation of collaborative filtering for recommender systems". International Journal of Advanced Computer Science and

Applications/International Journal of Advanced Computer Science & Applications, 12(3). https://doi.org/10.14569/ijacsa.2021.0120367.

[10] Kharita, M. K., Kumar, A., & Singh, P. (2018). "Item-Based Collaborative Filtering in Movie Recommendation in Real time". 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC). https://doi.org/10.1109/icsccc.2018.8703362.

Author Details

Shradha U. Lipane, M-Tech
Department of Computer Science and Engineering, CSMSS College of Polytechnic, Kanchanwadi, Chh.Sambhajinagr, India.

Research Interests: *Artificial Intelligence, Machine Learning, Deep Learning.*

Brief Bio: *Shradha U. Lipane is a Lecturer in the Department of Computer Science and Engineering at CSMSS College of Polytechnic. Her research focuses on machine learning applications.*

ORCID: 0009-0006-4584-7645

