ISSN 2063-5346

AN ANALYTICAL INSIGHT INTO DATASET PREPARATION AND ANALYSIS OF A PHYSICAL LIBRARY DATASET FOR A RECOMMENDER SYSTEM DESIGN

Monika Verma[a], Pawan Kumar Patnaik[b]

Article History: Received: 10.05.2023 Revised: 29.05.2023 Accepted: 09.06.2023

Abstract

The internet is abundant with information. But useful content, which caters to our particular requirements, is elusive. A recommender system (RS) is a sophisticated information filtering system that sorts through voluminous data and delivers results tailored to user preferences. The two standard RS techniques are collaborative filtering and content-based filtering. The proposed method, a hybrid model, utilized a real-time dataset from the Bhilai Institute of Technology, Durg, accounting for the number of times a book was issued plus the issue and return dates. A timestamp was used to assign weights to each book. Data sparsity was addressed using pre-processing techniques. Competitive prediction accuracy was obtained through user clustering and direct predictions. The overall accuracy of faculty transaction datasets was 98%.

Keywords*: hybrid recommender system; data pre-processing; dataset preparation; weighted hybridization; timestamp approach.*

[a]Department of Computer Science and Engineering, Bhilai Institute of Technology, Durg(C.G.), India

[b]Department of Computer Science and Engineering, Bhilai Institute of Technology, Durg(C.G.), India

Corresponding Author: monika04verma@gmail.com

Co-author: pawanpatnaik37@gmail.com

DOI:10.48047/ecb/2023.12.9.10

I. INTRODUCTION

A recommender system (RS) is a type of information filtering system that makes product recommendations to users based on their preferences or actions. E-commerce, social media, audio streaming, and movie streaming platforms all make frequent use of this system. To determine the products a user is most likely to be interested in, recommender systems employ many techniques. Recommender systems can increase user engagement, boost sales and enhance user experience on the platforms that use them [1]. However, they also raise privacy concerns because personalized recommendations require access to user data.

There are a number of different recommender systems, but the two primary kinds are content-based filtering and collaborative filtering. Other recommender system types also exist. The following is a concise explanation of each type:

Content-based filtering: A content-based recommender system suggests items to users based on the content of the items they have previously engaged with. For instance, if a user has viewed multiple science fiction movies, this type of recommender system would recommend additional science fiction movies to the user[2]. Content-based filtering uses item attributes, such as keywords or genres, to determine similarities between items and recommend similar items to users[3].

Collaborative filtering: This type of recommender system recommends items to users based on the preferences and behavior of similar users. For example, if User A has similar movie preferences as User B, a collaborative filtering recommender system would recommend movies that User B has enjoyed to User A[4]. Collaborative filtering uses data on user interactions with items to determine similarities between users and recommend items that similar users have enjoyed.

Hybrid recommender systems are designed to provide more precise recommendations by integrating both collaborative filtering and content-based approaches. Apart from these, there are other types of recommender systems, such as knowledge-based systems, demographic-based systems, and community-based systems[5].

Recently [6], some new recommender systems have been proposed and are sorted broadly as follows:

- Demographic
- Time aware
- Knowledge-based
- Community-based
- Semantic-based
- Context aware

However, designing an effective RS system can be challenging for several reasons. For example, data sparsity occurs when users rate only a small fraction of the available items. Cold start occurs when a new user or item has no rating history, which makes it difficult to accurately predict user preferences and leads to irrelevant recommendations. Scalability is a major concern, because as the number of users and items in the system grows, it becomes challenging to handle the sheer volume of data. Diverse recommendations, privacy concerns related to protecting users' personal information, trust and user satisfaction are also major challenges in designing a recommender system.

In our proposed approach, we aimed to recommend a book to the user (student) of the college physical library by easing their decision-making in choosing a book on a particular subject. An engineering college, for example, has various branches like computer science and engineering (CSE), electrical engineering, electronics engineering, civil engineering and mechanical engineering. Each branch has different subjects for its coursework, and each subject has books by different authors [7].

The association rule approach was used in this research, in accordance with book classifications, to identify the link between books individuals were interested in reading and their availability [8]. This technique of suggesting books could be useful to users in searching for books and also improve search outcomes. It not only increased the efficiency of the physical library but also helped lower its maintenance cost. Additionally, it assisted users in browsing the extensive range of books on display and may have also encouraged individuals to develop a greater interest in reading[9].

The timestamps of activities indicate the temporal characteristics of the user's choice. When compared to other context dimensions, time has the benefit of being easier to record, as almost every electronic device has a clock that can record the timestamp of an interaction. An alternate method used by several researchers is to include temporal information. In this method, we used timestamp data to improve the effectiveness of the book recommendation system in libraries. Timestamp information can be used for a variety of purposes:

(1) Some studies simply included time as a context, such as daytime, evening time or the weekend.

(2) As user preferences change over time and the user rating for a book tends to decrease, some studies used the rating's aging factor in recommending books.

Hybrid recommender systems have been demonstrated to perform better than single recommendation ones in many different applications. These systems are used extensively in business[10,11]. These systems are used extensively in business[12]. However, a hybrid recommender system can be difficult to design and execute without extensive knowledge of different recommendation methods. Some hybridization methods are not order sensitive to application: weighted, mixed, switching and feature combination. Other hybridization methods – cascade, feature augmentation and meta level – are sensitive to the sequence in which they are applied [13].

II. RELATED WORKS

Recommender systems process vast quantities of data to provide the active user with personalised recommendations. By tracking the user's online purchases and inquiries, they maintain and process information regarding various products/services and the user's preferences, expressed through ratings over time. Recommendation systems rely on implicitly or explicitly collected data. Exhaust data are by-products of user activity that may or may not be utilised.

We can collect explicit data through registration forms and profile information, online user reviews supplied by users [14–16]. Together, these data serve as the input for RS models that predict user preferences [17–19], although the accuracy of these models depends on data quality and volume. Regarding machine learning techniques for processing such data, the majority of current recommender systems are based on deep learning architectures that "provide a robust framework for supervised learning."

Focusing more specifically on post hoc data cleaning, there are many techniques in the research literature and many products in the marketplace. The space of techniques and products can be categorised fairly neatly by the types of data they target [20,21]. Data cleaning techniques are classified by data type. Quantitative data are whole numbers or floats that quantify quantities of interest. Categorical data are names or identifiers that organise data into groups or categories. Identifiers or keys are a subset of categorical data that uniquely identify objects or properties[22].

Preprocessing is also an important component of a typical text classification framework. The evaluation was conducted using all conceivable combinations of the

preprocessing tasks, accounting for several factors including precision, domain, language and dimension reduction [23,24]. Extensive experimental analysis revealed that appropriate combinations of preprocessing tasks significantly enhanced classification accuracy, while inappropriate combinations degraded it. Consequently, the preprocessing phase is as crucial to text classification as feature extraction, feature selection and classification[25].

A large dataset contains a several well-organised rows and columns. However, this is untrue of data, which typically arrives as text documents and unstructured datasets. In the real world, information originating from an application is typically noisy, contains some errors or outliers, is incomplete, lacks attribute values and is inconsistent, consisting of code or name discrepancies. Data pre-processing is a data mining technique that transforms unprocessed data into an understandable format and is a crucial stage in the machine learning process, which cannot comprehend unstructured text-based datasets[26,27].

III. PROBLEM IDENTIFICATION

When attempting any sort of data mining, it is crucial to remember that data integrity is essential. Nearly 80% of extraction efforts are frequently spent on enhancing data quality. It is possible for the information gleaned from the records to be unreliable, noisy and lacking in detail. Good data will have the following characteristics: accurate, comprehensive, consistent, timely, credible, interpretable and accessible. Pre-processing data is necessary to give it the aforementioned qualities and make it amenable to knowledge mining. The issue/return transactional database of the library, containing an enormous quantity of content and books that cannot be readily located by users – primarily, student communities – is the root of the book recommender system's design problems. It is not easy for readers to track down authoritative works on the specific subjects that pique their interest. Even if they do find the popular books, it is still a hassle to track down another book (item) in the same subject area; additionally, the chosen book as a whole is not available for an issue transaction. The necessity to solve this issue has inspired this research effort[28].

IV. DATASET DESCRIPTION

In this research, the book recommendation dataset was taken from the library book (item) and issue/return transaction databases. This dataset consisted of 50,152 transactions rows of five years, where users had been issued around 14,000 books (items). The data was stored as integer, varchar and date/time, in rows and columns. Further details about the dataset descriptions along with the attributes are given in Table I. The database consisted of three entities: book details (item), book demographic attributes and issue/return transactions. Book details consisted of the following attributes: account number, author, title, publication, edition and number of copies as well as volume price status. Book demographic consisted of the following attributes: name, branch, subject, author name, text/reference book tag and number of copies. Issue/return transactions consisted of the following attributes: member ID, member name, item number, title, accession number, issue date for the books, and return date for the books. The proposed method efficiently provided recommended books with more accuracy. Demographic data was provided based on book tag (text/reference), issue frequency, issue span. The attributes of the transaction database are listed below.

A. Dataset Distribution

The transaction dataset contains two types of information: faculty transactions and student transactions from the different branches of an engineering college. First, faculty and student data are segregated using their user ID in a separate file. The students' transactions will help us train the model and get recommendations, while faculty transactions will help us enhance the recommendations by selecting the most

appropriate book for the particular subject of the concerned branch. The dataset distribution of student transactions is shown in Fig. 1.

Fig. 1 shows the distribution of the book based on branches such as civil, CSE, electrical engineering, electronics engineering and mechanical engineering and subsequently by their types – that is, text (T), reference (R) and miscellaneous (M). Tagging of book type has been done explicitly by the faculty member of the concerned branch. The subject of the book has also been tagged for demographic purposes.

TABLE I

ATTRIBUTE ANALYSIS OF THE COLLEGE PHYSICAL LIBRARY DATABASE

Book	**Demographic Attributes**	**Transaction database**
Item number	Book-Tag (Text/Reference)	Member ID
Book Title	Issue Frequency	Member Name
Accession Number	Issue Span	Item number
Author		Title
Publication		Accession Number
Pages		Issue Date for the books
Number of copies		Returning Date for the books
Amount		
Branch		
Subject		

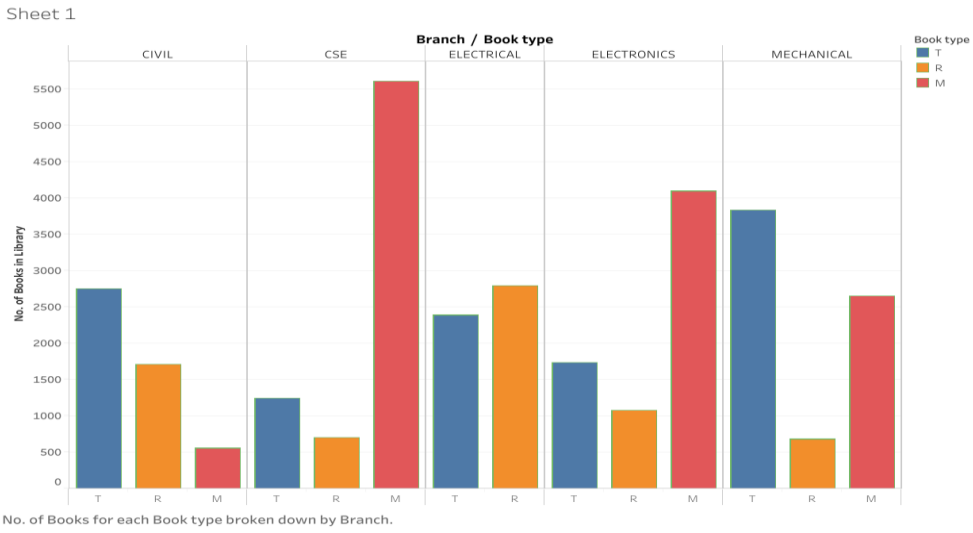

Fig. 1 Branch-wise Distribution of various Book Types

B. Mapping Of Books into Transaction Data

The entire transaction sample has been mapped with the book item information table to include the subject, book type and total number of copies of that book available in the library, which will enhance the demographic features of every item (book) in the transaction database. Issue

frequency count has been taken as an implicit rating for applying CF techniques.

V. DATA PRE-PROCESSING

Data pre-processing is essential to data mining, as it involves the preparation and transformation of data into a form suitable for mining. The process seeks to reduce data size, identify relationships between data, normalize data, remove outliers, and extract features from data. It consists of data cleansing, feature selection, missing value imputation and data normalization. **Figure 2** shows the steps followed in data pre-processing.

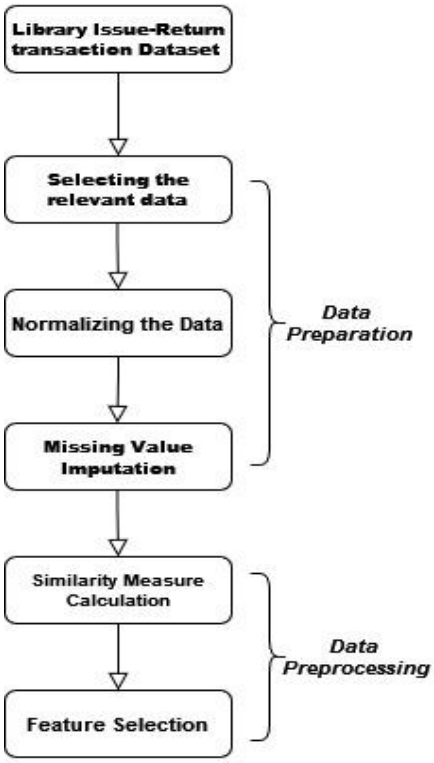

Fig. 2 Data Pre-processing Steps

A. Data Cleaning

The first stage in data pre-processing is data cleansing. The process involves eradicating all incomplete records, noise values, outliers and inconsistent data useless for mining, as they will influence the mining procedure and result in unreliable and substandard output.

B. Missing Values

If there are unrecorded values in any columns or rows, the following can be done: ignore the -tuples; manually fill the missing values; use a global or particular constant, the mean value of an attribute, the mean value of an attribute of a particular class or the most probable value to fill the missing values. In our case, we manually filled in the missing values and sometimes ignored the -tuple completely.

C. Feature Selection

In machine learning, feature selection methods have been used for long to identify the most pertinent features for classification or to eliminate noisy or redundant features. When redundant or chaotic features are eliminated, simpler, more generalizable models can be trained and evaluated in less time. For our dataset, we selected only the relevant features required to provide accurate recommendations.

D. *Timestamp-based similarity measure*

The Pearson correlation or other correlation-based similarities are used to calculate the similarity $S(c,d)$ between two users, c and d, or $S(e,i)$ between two things, e and i. The two most often used similarity metrics are cosine based and correlation based. Here, similarity was calculated between the department and the book using the Pearson correlation coefficient (PCC). PCC, which is used to assess how much two variables are linearly related to one another, is defined as follows:

$$S(c,d)=\frac{\sum_{e\in E}(r_{c,e}-r_c^{'})(r_{d,e}-r_d^{'})}{\sqrt{\sum_{e\in E}(r_{c,e}-r_c^{'})^2}\sqrt{\sum_{e\in E}(r_{d,e}-r_d^{'})^2}} \quad (1)$$

In equation (1), $r_{c,e}$ represents the client c 's rating of item e and $r_{d,e}$ represents user d 's rating for that exact item. The total number of products available is m, and the average ratings for item e are r_c and r_d. The measure's similarity, however, is unable to account for the different actions that people take while rating the same thing.

Some other correlation-based similarities are constrained Pearson correlation, which is a variation of the Pearson correlation that

uses the midpoint rather than the mean rate; Spearman rank correlation, which is similar to the Pearson correlation but uses ranks instead of ratings; and Kendall's correlation, which is similar to the Spearman rank correlation but only uses relative ranks to calculate the correlation.

VI. PROPOSED MODEL

In the proposed system, each department user receives personalised book recommendations based on a hybrid RS. The ranking-based system of recommendations made the system more objective. The hybrid RS increased precision and scalability. The input dataset was initially pre-processed, and weights were assigned to each book attribute. Then, using the proposed hybrid RS, they were categorised for each department. The estimated score was then utilised to generate a list of recommendations.

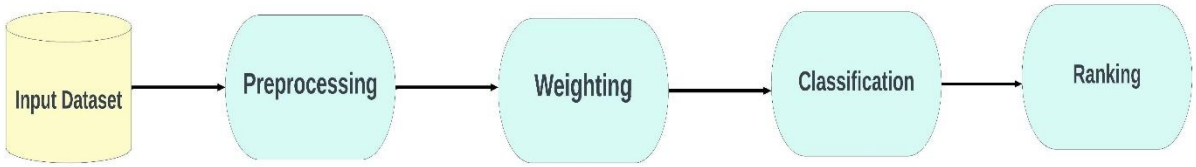

Fig.3 Workflow of the proposed method

The proposed method is illustrated in **Figure 3**. The initial step involved labelling the input dataset with the department name as the descriptor. The tagging helped reduce the processing complexity required to deal with the particular data. Hence, the effectiveness of the RS was enhanced. By using a pre-processing strategy, it was possible to increase the accuracy of the RS. After the pre-processing step, the Timestamp Pearson Correlation Coefficient provided a weight for each book based on the timestamp, which included information such as the number of times a book was issued, the day the book was issued and the date it was returned. Hidden Markov Discriminant Analysis, which is a Content-Based Filtering system, and Weighted Fuzzy Ranking, which is a CF system, can both be included in the concept of the hybrid RS suggested here. In the end, a list of books to recommend to every department was compiled using the scores that each book received after going through the fuzzy ranking process. The candidate can select books as per their interest with the help of this excellent recommendation list.

VII. RESULT AND DISCUSSIONS

The proposed approach for an RS was implemented on a Python platform and compared to extant approaches based on performance metrics such as precision, mean absolute error (MAE), kappa score, and run time.

A. Performance Comparison with conventional techniques

Figure 4 gives graphical representations of the performance attained by the proposed method. To demonstrate the effectiveness of the proposed method, conventional methods such as the hidden Markov model (HMM), linear discriminant analysis (LDA), CF, and CBF are compared.

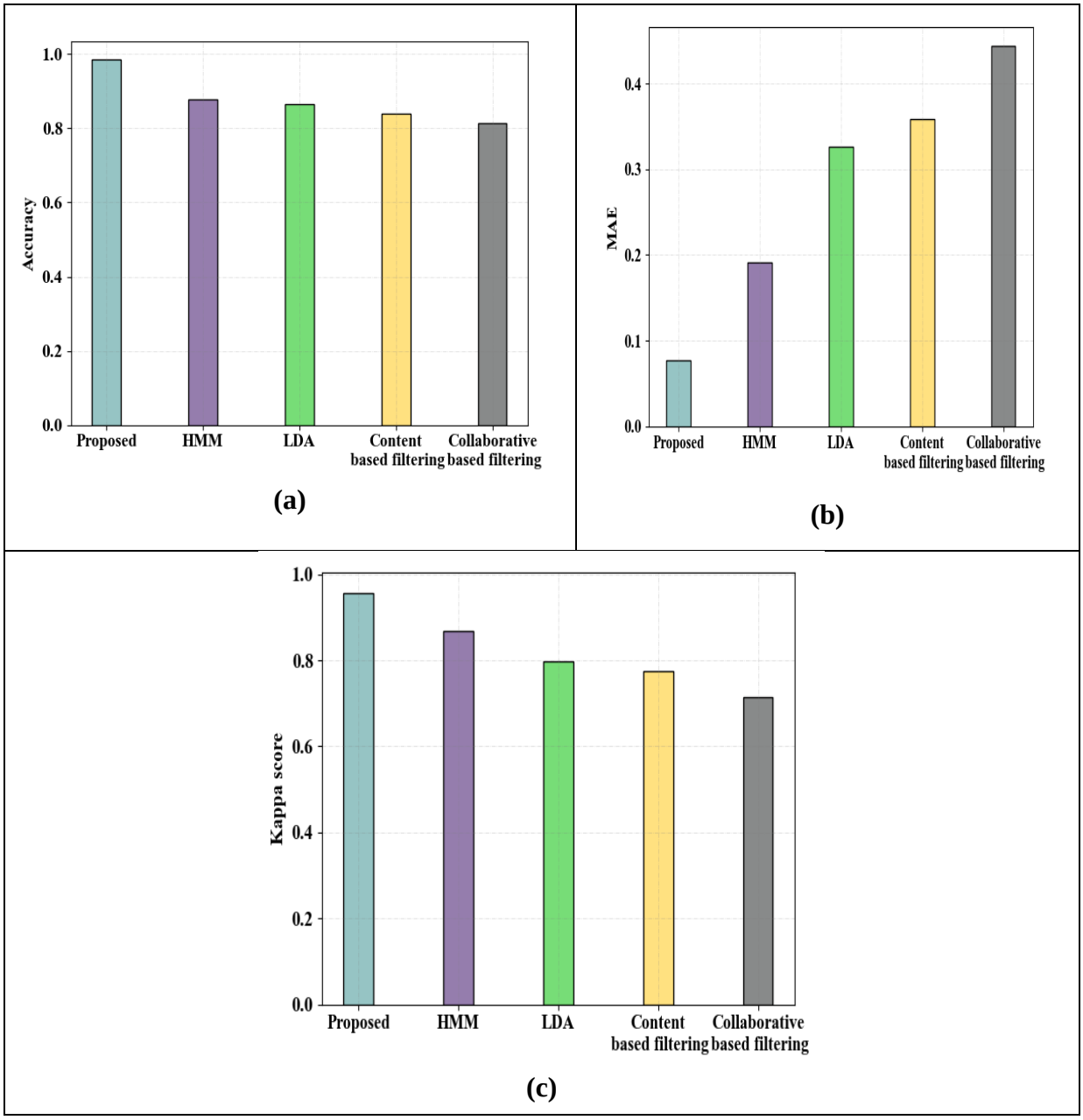

Fig.4 Performance accuracy for the faculty dataset (a) Recommendation accuracy(b) MAE analysis (c) Kappa performance

B. Ranking of Books Department Wise

This section does an analysis on a monthly basis for the faculty dataset, focusing on the books that were ranked. The proposed WFR approach provided an accurate ranking of the books based on the score that was generated through the fuzzy ranking. The following figure provides in-depth insights into the simulation's results:

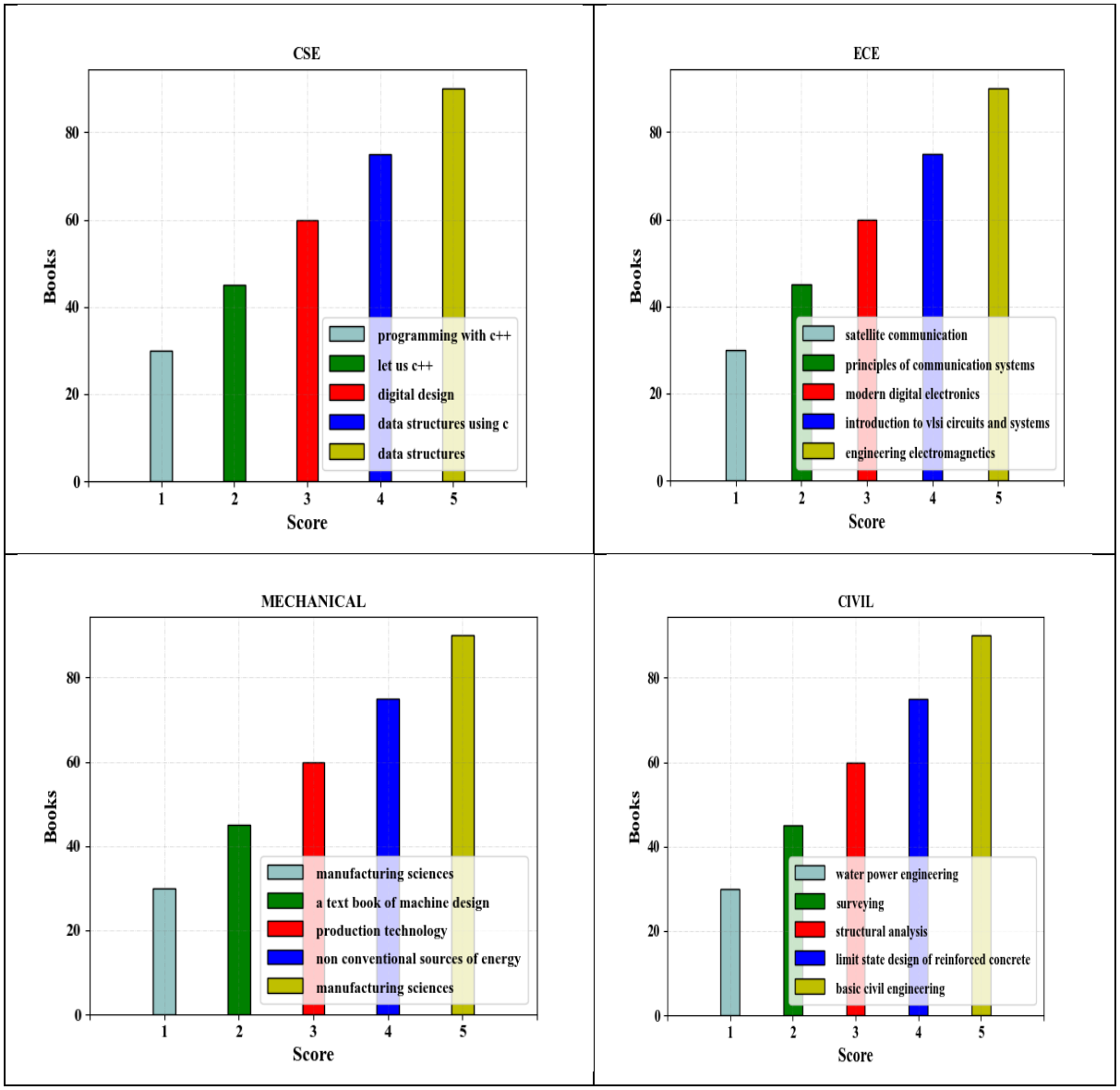

Fig. 5 Month wise ranking of recommended books for all departments from the faculty dataset

C. Performance Comparison with Recent Developments

We compared our proposed model with existing techniques. The performance measures used are accuracy, kappa score and MAE to evaluate the efficacy of our algorithm against existing ones. Table 2 tabulates the performance obtained by existing techniques. From the table, it is clear that the proposed method delivers better performance in comparison. The existing methods are time-consuming and have low accuracy due to poor feature learning ability and increased over-fitting issues.

TABLE II

PERFORMANCE OBTAINED BY EXISTING TECHNIQUES FOR FACULTY TRANSACTION DATASET

Performance measure	**SVR**[29]	**LSTM**[30]	**AE**[31]	**Proposed**
Accuracy	94%	93%	94%	**98%**
Kappa	81.6%	78.9%	86.5%	**97.7%**
MAE	0.28	0.32	0.26	**0.077**

CONCLUSION

The component of the information filtering system known as the recommendation system is responsible for assisting with the prediction of the user's preference of any given item. The primary objective of this research is to make recommendations for which books should be included in a personalized library in order to guide students in selecting suitable books from the various subject areas. At the beginning of the process, labelling is carried out for each department, and after that, timestamp-based weighting is carried out for each book by making use of the number of days that are estimated between issue date and the return date and the number of times the book is issued. After that, PCC is carried out in order to conduct an analysis of the differences between the departments and the books that relate to them correspondingly. In conclusion, the HMDA-WFR technique is emphasized so that the book can be appropriately classified and ranked according to the many departments. Python is utilized as the platform for carrying out the construction of the suggested recommendation system, and a real-time dataset sourced from the Bhilai Institute of Technology in Durg is employed. In the case of the faculty, the proposed method achieves an overall accuracy of 98% in the experimental setting.

ACKNOWLEDGMENTS

We would like to express our sincere gratitude to the management of Bhilai Institute of Technology, Durg for sharing the physical library dataset for research work, as well as the Department of Computer Science and Engineering for providing the laboratories to accomplish the practical part of the research.

*Conflict of Interest***:** The authors declare that they have no conflict of interest

REFERENCES

[1] D.H. Park, H.K. Kim, I.Y. Choi, J.K. Kim, “A literature review and classification of recommender systems research”, *Expert Syst Appl.* 39 (2012) 10059–10072. https://doi.org/10.1016/j.eswa.2012.02.038.

[2] R.J. Mooney, L. Roy, “Content-Based Book Recommending Using Learning for Text Categorization”, (1999). https://doi.org/10.1145/336597.336662.

[3] M.J. Pazzani, D. Billsus, “Content-based recommendation systems”, *The Adaptive {W}eb: {M}ethods and Strategies of {W}eb Personalization*. (2007) 325–341.

[4] T. Bogers, A. Van Den Bosch, “Collaborative and content-based filtering for item recommendation on social bookmarking websites”, 2009.

https://doi.org/10.1007/978-0-387-85820-3.

[5] J. Bobadilla, F. Ortega, A. Hernando, A. Gutiérrez, "Recommender systems survey", *Knowl Based Syst.* 46 (2013) 109–132. https://doi.org/10.1016/j.knosys.2013.03.012.

[6] R. Burke, A. Felfernig, M.H. Göker, "Recommender Systems : An Overview", *AI Mag*. (2011) 13–18. https://doi.org/10.1609/aimag.v32i3.2361.

[7] M. Verma, A. Rawal, "An Enhanced Item-Based Collaborative Filtering Approach for Book Recommender System Design", *ECS Trans.* 107 (2022) 15439–15449. https://doi.org/10.1149/10701.15439ecst.

[8] A.A. Kardan, M. Ebrahimi, "A novel approach to hybrid recommendation systems based on association rules mining for content recommendation in asynchronous discussion groups", *Inf Sci (N Y)*. 219 (2013) 93–110. https://doi.org/10.1016/j.ins.2012.07.011.

[9] M. Chandak, S. Girase, D. Mukhopadhyay, "Introducing Hybrid Technique for Optimization of Book Recommender System", *Procedia - Procedia Computer Science.* 45 (2015) 23–31. https://doi.org/10.1016/j.procs.2015.03.075.

[10] T. Srikanth, M. Shashi, "An effective preprocessing algorithm for model building in collaborative filtering-based recommender system", 2019.

[11] S. Bhaskaran, R. Marappan, "Design and analysis of an efficient machine learning based hybrid recommendation system with enhanced density-based spatial clustering for digital e-learning applications", *Complex and Intelligent Systems*. (2021). https://doi.org/10.1007/s40747-021-00509-4.

[12] M. Verma, A. Rawal, C. Science, C. Science, "An Analytical Insight to Hybrid Recommender Systems", 8 (2018) 331–338.

[13] R. Burke, "Hybrid Recommender Systems : Survey and, User Model" *User-Adapt Interact.* 12 (2002) 331–370. https://doi.org/10.1023/A:1021240730564].

[14] K. jae Kim, H. Ahn, "A recommender system using GA K-means clustering in an online shopping market", *Expert Syst Appl*. 34 (2008) 1200–1209. https://doi.org/10.1016/j.eswa.2006.12.025.

[15] T.C. Huang, Y.M. Huang, "Where are my cooperative learning companions: designing an intelligent recommendation mechanism", *Multimed Tools Appl.* 76 (2017) 11547–11565. https://doi.org/10.1007/s11042-015-2678-2.

[16] M.Y.H. Al-Shamri, K.K. Bharadwaj, "Fuzzy-genetic approach to recommender systems based on a novel hybrid user model", *Expert Syst Appl.* 35 (2008) 1386–1399. https://doi.org/10.1016/j.eswa.2007.08.016.

[17] T. Horváth, A.C.P.L.F. de Carvalho, "Evolutionary computing in recommender systems: a review of recent research", *Nat Comput.* 16 (2017) 441–462. https://doi.org/10.1007/s11047-016-9540-y.

[18] Review_of_Data_Preprocessing_Techniques, (n.d.).

[19] N.W. Rahayu, R. Ferdiana, S.S. Kusumawardani, "A systematic review of learning path recommender systems", *Educ Inf Technol (Dordr).* (2022). https://doi.org/10.1007/s10639-022-11460-3.

[20] F. Kamiran, T. Calders, "Data preprocessing techniques for classification without discrimination", *Knowl Inf Syst.* 33 (2012) 1–33. https://doi.org/10.1007/s10115-011-0463-8.

[21] S. Tyagi, N.K. Tyagi, A.K. Solanki, "An Algorithmic Approach to Data Preprocessing in Web Usage Mining", 2010. https://www.researchgate.net/publication/228617684.

[22] J.M. Hellerstein, "Quantitative Data Cleaning for Large Databases", 2008. http://db.cs.berkeley.edu/jmh.

[23] H.J. Oudah, M.H. Hussein, "Exploiting social trust via weighted voting strategy for recommendation systems improvement", Karbala International Journal of Modern Science. 9 (2023). https://doi.org/10.33640/2405-609X.3295.

[24] A.S. Al-Fallooji, A. Al-Azawei, "Predicting Users' Personality on Social Media: A Comparative Study of Different Machine Learning Techniques", *Karbala International Journal of Modern Science*. 8 (2022) 617–630. https://doi.org/10.33640/2405-609X.3262.

[25] A.K. Uysal, S. Gunal, "The impact of preprocessing on text classification", *Inf Process Manag.* 50 (2014) 104–112. https://doi.org/10.1016/j.ipm.2013.08.006.

[26] M. Basil, A. Ahmad, M. Altamimi, M.B. Albayati, A.M. Altamimi, "Analyzing COVID-19 Vaccine Adverse Reactions Using Machine Learning", (2022). https://doi.org/10.33640/2405-609X.3271.

[27] S. Bhaskaran, R. Marappan, "Enhanced personalized recommendation system for machine learning public datasets: generalized modelling", *simulation, significant results and analysis, International Journal of Information Technology (Singapore*). (2023). https://doi.org/10.1007/s41870-023-01165-2.

[28] D. Sarma, T. Mittra, M.S. Hossain, "Personalized Book Recommendation System using Machine Learning Algorithm", n.d. www.ijacsa.thesai.org.

[29] B. Gopikrishna, S. Ashwini, "A Peculiar Approach for Hotel Recommendation System using SVR Algorithm Over Matrix Decomposition for Improved Accuracy", *in: Proceedings - 2022 6th International Conference on Intelligent Computing and Control Systems, ICICCS 2022, Institute of Electrical and Electronics Engineers Inc.*, 2022: pp. 348–351. https://doi.org/10.1109/ICICCS53718.2022.9788182.

[30] G. Spoorthy, S.G. Sanjeevi, "Multi-criteria– Recommendations using Autoencoder and Deep Neural Networks with Weight Optimization using Firefly Algorithm", *International Journal of Engineering Transactions C: Aspects.* 36 (2023) 130–138. https://doi.org/10.5829/ije.2023.36.01a.15.

[31] R. Shen, "A Recommender System Integrating Long Short-Term Memory and Latent Factor", *Arab J Sci Eng.* 47 (2022) 9931–9941. https://doi.org/10.1007/s13369-021-05933-9.