University Recommendation System for Undergraduate Studies in Bangladesh Using Distributed Machine Learning

Ahmed Nur Merag[a], Rezwana Chaudhury Raka, Sumya Afroj, Md Humaion Kabir Mehedi[b] and Annajiat Alim Rasel

Department of Computer Science and Engineering, Brac University, 66 Mohakhali, Dhaka 1212, Bangladesh

Keywords: University Recommendation System, Popularity Based Recommender Model, Collaborative Filtering, KNN, SVD, Cosine Similarity, Pearson's Correlation, Data Mining, Distributed Machine Learning.

Abstract: The study proposes a distributed machine learning-based university recommendation system (URS) in Bangladesh to help undergraduate students make informed decisions based on user ratings. The system uses advanced distributed machine learning models such as collaborative filtering and popularity-based recommender model which consists of KNNwithmeans model and singular value decomposition (SVD) model to process data and provide accurate recommendations, significantly enhancing the university selection process for students. This study advances educational technology and provides a useful tool for undergraduates in Bangladesh.

1 INTRODUCTION

Bangladesh, a country famous for its rich cultural legacy and expanding academic prospects, has seen an increase in the number of universities and academic programs catering to a wide range of fields. With the higher education sector developing, the need for intelligent and data-driven techniques to assisting students in finding the most appropriate university and program becomes critical. This research paper includes a thorough investigation and development of a URS for undergraduate studies in Bangladesh.

The potential for recommendation systems to change higher education in Bangladesh is great. Universities can improve student experiences by utilizing data-driven algorithms, which can help with course selection, career counseling, student engagement, support services, and alumni networking. This fusion of technology and education creates a unique, student-centered environment that supports success and brings satisfaction. By combining distributed machine learning and recommendation algorithms, distributed machine learning highlights this paradigm and produces scalable, tailored student suggestions. In order to improve students educational experiences, this project sets out to build a university recommendation system for undergraduate studies in Bangladesh. The fundamental goal of this research is to use two separate recommendation models, the popularity based recommender model and the collaborative filtering recommender model. Furthermore, the examination includes advanced strategies such as the KNN with means model and the SVD - model based collaborative filtering. These models work together to provide a strong framework that uses data analytics and machine learning to provide university recommendations to aspiring undergraduates. We have not provide the actual names of the universities rather used a unique nameId to identify them discretely.

This study is organized as follows: The earlier research utilizing distributed machine learning and recommendation algorithms are discussed in section 2. A short description of the dataset is given in Section 3. Section 4 includes the methodology and section 5 contains performance analysis, accuracy of implemented models and results. Lastly, concluding remarks are given in section 6.

2 LITERATURE SURVEY

Li et al. propose a distributed collaborative filtering algorithm using MapReduce for improved performance and scalability in massive recommenda-

[a] https://orcid.org/0009-0004-9248-4010
[b] https://orcid.org/0000-0002-5759-022X

Merag, A., Raka, R., Afroj, S., Mehedi, M. and Rasel, A.
University Recommendation System for Undergraduate Studies in Bangladesh Using Distributed Machine Learning.
DOI: 10.5220/0012256100003543
In *Proceedings of the 20th International Conference on Informatics in Control, Automation and Robotics (ICINCO 2023) - Volume 1*, pages 352-358
ISBN: 978-989-758-670-5; ISSN: 2184-2809

tion systems (Li et al., 2014). The authors developed a distributed matrix factorization method for recommender systems using Apache Mahout, addressing challenges in managing collaborative filtering tasks in university recommendation systems (Ghoting et al., 2011). The researchers showcase MLlib, a distributed machine learning library for Apache Spark, for academic institutions' recommendation systems, benefiting Bangladeshi algorithms development (Meng et al., 2016). They suggest a distributed learning framework for personalized recommendation systems, utilizing data segmentation and parameter synchronization for scalability and communication reduction (Zhang et al., 2019b). The researchers present federated learning for privacy-protected university recommendation systems, focusing on design, scalability, and collaborative filtering (Bonawitz et al., 2019). The authors put forward a federated collaborative filtering approach for personalized suggestions, addressing privacy concerns and demonstrating potential in university recommendation systems(Yang et al., 2020). They introduce a hybrid recommendation system combining collaborative and content-based filtering in a distributed setting, improving university suggestions quality in Bangladesh(Jin et al., 2017). The authors recommend a hybrid distributed recommendation system combining deep learning and collaborative filtering for improved undergraduate studies in Bangladesh (Yao et al., 2019). The academics discuss scalability in distributed machine learning, offering insights for algorithm development. They emphasize the importance of scalability in recommendation systems for Bangladeshi institutions, accommodating large user bases and data volumes (Kumar and Kaur, 2020). The investigators put forward real-time clustering methods for Bangladeshi university's recommender systems (Islam et al., 2019). The scholars recommend distributed deep learning methods for massive recommender systems, focusing on using distributed computing frameworks to train models, enhancing functionality and scalability in academic institutions' recommendation systems (Choi et al., 2018). The analysts advocate a distributed learning framework for large-scale heterogeneous collaborative filtering in their 2019 paper, addressing scalability and data sources issues in recommendation systems. The paper offers valuable insights into managing diverse data in academic recommendation systems (Zhang et al., 2019a). The investigators develop a distributed recommendation system using LightFM algorithm for e-commerce, applicable to university recommendation systems (Wang et al., 2020). The researchers explore distributed machine learning for predictive customization in university recommendation systems, without direct connection to colleges (Yu et al., 2020). They analyze hybrid recommendation systems using collaborative filtering and deep learning, providing insights for researchers in Bangladesh. This review study is useful for developing hybrid systems for undergraduate courses (Kaur and Chawla, 2020).

3 DATASET DESCRIPTION

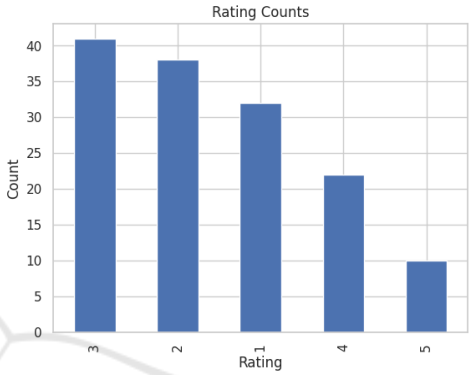

Figure 1: Rating Distribution from 1 to 5.

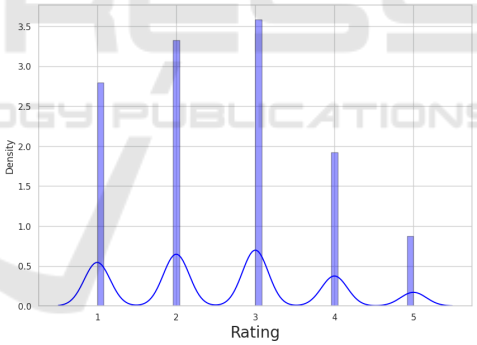

Figure 2: Rating Distribution-II.

The dataset we have used in this research paper is created from scratch. The file contains features like unique userId, university names and ratings collected from various social media sites which were given out of 5 (five) stars by the users. Every user is identified with a unique id here. Furthermore, ratings of 143 universities are given in this dataset as features. From figure 1, we see that the most frequent rating is 3 and the least frequent rating is 5. Here, 1 is the lowest rating and 5 is the highest. In figure 2, we get a clear visual idea about the density of the ratings. Ratings may change due to users opinion. In other words, these ratings are optimized to conduct the research smoothly.

4 METHODOLOGY

This research starts with collecting data which is described in the dataset description section and then we move on to data pre-processing and feature engineering. After that, data analysis and visualization is done. Consequently, training and testing, model selection and finally evaluation of the selected models and analysis of the generated results are performed. From figure 3, we get a clear idea about the architecture of this study.

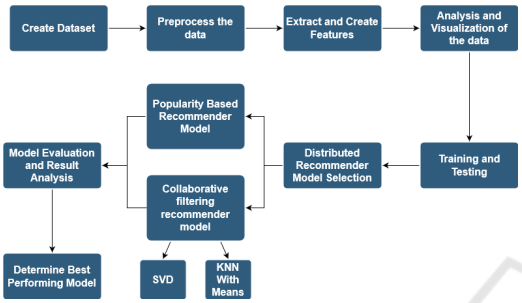

Figure 3: Architectural Schematics.

4.1 Data Pre-Processing

Data pre-processing means cleaning and processing the collected data. In this step handling missing values, removing noise, and transforming the data into a applicable format for the recommendation models which is incorporated in the URS for undergraduate studies in bangladesh is done. Also, basic exploratory data analysis is performed.

4.2 Feature Engineering

Now, we extract important characteristics from the pre-processed data. This entails encoding categorical variables, normalizing numerical features, and creating new features that capture important aspects of the data.

4.3 Data Analysis and Visualization

The process of obtaining insights and knowledge from data by utilizing various analytical approaches and presenting the results in visual formats is referred to as data analysis and visualization. It means analyzing and interpreting data in order to uncover patterns, trends, correlations, and other useful information that can improve decision-making and problem-solving.

We generated rating distribution, top rating count distribution grouped by the names of universities, top rating count distribution grouped by the user ids, mean rating distribution grouped by the names of the universities, mean rating - rating count distribution grouped by the names of the universities, mean rating distribution grouped by user ids and mean rating - rating count distribution grouped by user Ids.

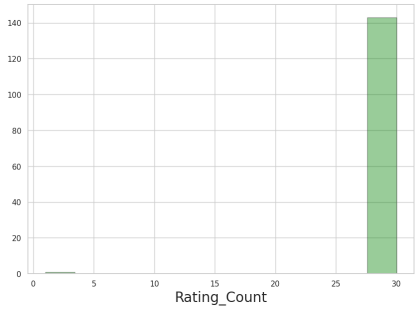

Figure 4: Top Rating Count Distribution grouped by names.

Firstly, top rating count distribution grouped by the name of universities graph is generated and shown below in figure 4.

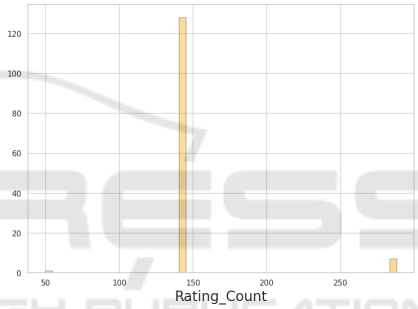

Figure 5: Top Rating Count Distribution grouped by user Ids.

Then, in figure 5, top rating count distribution grouped by user Ids is portrayed.

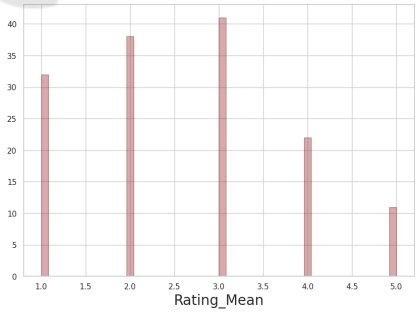

Figure 6: Mean Rating Distribution grouped by names.

After that, mean rating distribution grouped by university names is constructed as shown in figure 6.

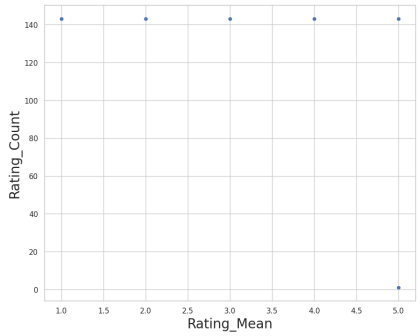

Figure 7: Mean Rating - Rating Count Distribution grouped by names.

Similarly, in figure 7, mean rating - rating count distribution grouped by the names of the universities is generated and shown below.

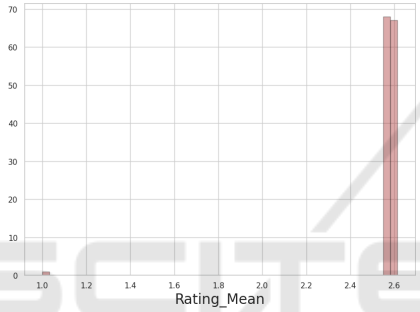

Figure 8: Mean Rating Distribution grouped by User ids.

Subsequently, mean rating distribution grouped by user ids is formed and shown in figure 8.

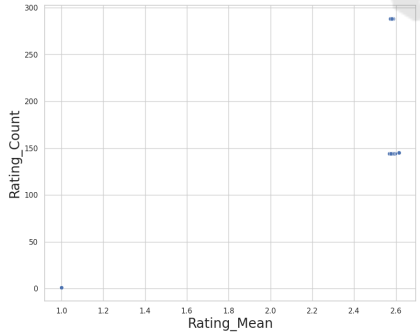

Figure 9: Mean Rating - Rating Count Distribution grouped by User Ids.

Finally, in figure 9, mean rating - rating count distribution grouped by user Ids graph is shown.

4.4 Training and Testing

We have divided the data into training and test sets in a 70:30 ratio.

4.5 Model Selection

We have selected 2 (two) appropriate recommendation models for this study. They are popularity based recommender model and collaborative filtering recommender Model. The Collaborative filtering recommender model also includes KNN With means model and SVD - model based collaborative filtering.

4.6 Popularity Based Recommender Model

Popularity-based recommender model promotes products based on popularity in a dataset. Algorithm replaces popular items with personalized recommendations, assuming interest among large groups without considering individual choices or activities.

Popularity Score = Number of Interactions (Likes, Views, etc.) for Item / Total Number of Users

4.7 Collaborative Filtering Recommender Model

The collaborative filtering recommender model uses user-item interactions to predict user preferences, enabling customized recommendations based on past agreement and future user interactions.

4.7.1 KNN With Means Model

KNN with Means predicts outcomes by estimating similarity between users or items, using previous interactions and calculating K-nearest neighbors based on individual ratings.When utilizing KNN with Means, the following equation can be used to forecast user 'u's rating of item 'i':

$$\widehat{r_{u,i}} = \mu_u + \frac{\sum v \in Nuk(i) similarity(u,v) \cdot (rv,i - \mu v)}{\sum v \in Nuk(i) similarity(u,v)}$$

Figure 10: KNN With Means Formula.

4.7.2 Singular Value Decomposition - SVD

Singular Value Decomposition (SVD) breaks user-item interaction user's matrix, singular value, and item matrices, capturing latent components and generating recommendations by finding lower-dimensional representations. The following is the SVD prediction formula for user 'u' and item 'i':

$$\widehat{r_{u,i}} = \mu + bu + bi + \sum_{f=1}^{k} Pu,f \cdot Qi,f$$

Figure 11: SVD Formula.

KNN with Means and SVD are collaborative filtering methods with strengths in local patterns and global patterns.

5 MODEL EVALUATION AND RESULT ANALYSIS

At this stage, we take a closer look at the recommendations and assess the performance of the trained models using appropriate evaluation metrics. This step helps us to understand how well the models are performing. We have not published the actual names of the universities, but rather a unique name Id is used to identify them discretely. Table 1 is given as an example below.

Table 1: Using nameId instead of actual name.

userId	nameId	University Name
R7S8T9U0	5947683210	BRAC University

Starting with popularity based recommender model, we try recommendation for three randomly picked users which are 'A1B2C3D4', 'E5F6G7H8' and 'M3N4O5P6'. The university names that are currently popular are used in the recommendation system based on popularity. It ranks the names depending on their popularity, i.e. the number of ratings. If an institution is highly regarded, It is more likely to be scored higher and thus recommended. Because it is dependent on the popularity of the name, it can not be individualized. Hence, the same list of names will be recommended to all users.

At first, we have generated the recommendation table for user 'A1B2C3D4' as shown in table 2.

Table 2: Recommendation table for user 'A1B2C3D4'.

sl num	user_id	nameId	score	Rank
110	A1B2C3D4	7913620854	113	3.0
141	A1B2C3D4	9826301574	110	7.0
32	A1B2C3D4	3106947285	107	14.0
115	A1B2C3D4	8192574306	107	16.0
18	A1B2C3D4	2356891470	106	19.0

Additionally, a score vs rank graph is also generated for the user 'A1B2C3D4' as shown in figure 12.

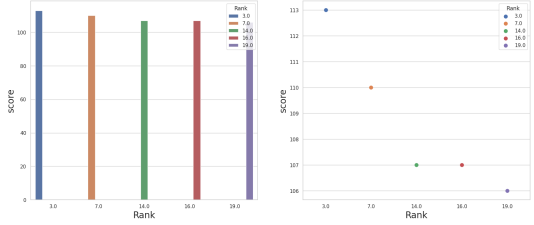

Figure 12: Score vs Rank graph for user 'A1B2C3D4'.

For user 'E5F6G7H8', a recommendation table which is table 3 and a similar graph which is figure 13 is also generated.

Table 3: Recommendation table for user 'E5F6G7H8'.

sl num	user_id	nameId	score	Rank
110	E5F6G7H8	7913620854	113	3.0
124	E5F6G7H8	8642097531	112	5.0
137	E5F6G7H8	9617530428	109	12.0
59	E5F6G7H8	4952386170	107	15.0
115	E5F6G7H8	8192574306	107	16.0

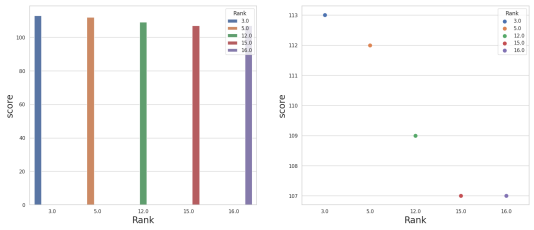

Figure 13: Score vs Rank graph for user 'E5F6G7H8'.

Finally, for the third user 'M3N4O5P6' a recommendation table is shown in table 4 along with a score vs rank graph in figure 14.

Table 4: Recommendation table for user 'M3N4O5P6'.

sl num	user_id	nameId	score	Rank
28	M3N4O5P6	2937501846	113	2.0
26	M3N4O5P6	2804175963	112	4.0
124	M3N4O5P6	8642097531	112	5.0
137	M3N4O5P6	9617530428	109	12.0
22	M3N4O5P6	2631894750	108	13.0

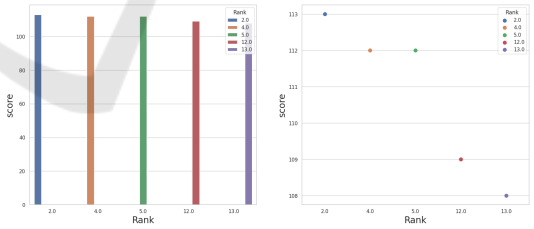

Figure 14: Score vs Rank graph for user 'M3N4O5P6'.

Because this is a popularity-based recommender model, we are getting a collection of names that are nearly identical for all three users. This model has an evaluation score of 2.84.

Now, we apply collaborative filtering recommender model which consists of KNN With means and SVD. KNN With means - memory based collaborative filtering is done first.

By using user-user collaborative filtering method, top 5 names are recommended for user 'A1B2C3D4' and displayed in table 5.

Table 5: Recommendation for ‘A1B2C3D4’ using KNN.

sl num	userId	nameId	Rating
0	A1B2C3D4	9876543210	5
1	A1B2C3D4	1234567890	5
2	A1B2C3D4	8192736450	5
3	A1B2C3D4	6512390847	5
4	A1B2C3D4	4728901563	5

Consequently, top 5 names are recommended for user ‘E5F6G7H8’ and portrayed in table 6.

Table 6: Recommendation for ‘E5F6G7H8’ using KNN.

sl num	userId	nameId	Rating
0	E5F6G7H8	2468135790	5
1	E5F6G7H8	1357924680	5
2	E5F6G7H8	9876543210	5
3	E5F6G7H8	1234567890	5
4	E5F6G7H8	5827369140	5

Lastly, top 5 names are recommended for ‘M3N4O5P6’ which is displayed in table 7.

Table 7: Recommendation for ‘M3N4O5P6’ using KNN.

sl num	userId	nameId	Rating
0	M3N4O5P6	2468135790	5
1	M3N4O5P6	1796253408	5
2	M3N4O5P6	7946832105	5
3	M3N4O5P6	6512390847	5
4	M3N4O5P6	5193640278	5

KNN (K-Nearest Neighbours) With means model has an test rmse value of 0.0727 and cross validation rmse value of 0.0843.

Regarding recommendations, each user will have unique names suggested to them according to the ratings given by comparable users. To locate comparable users, the KNN model used cosine similarity along with pearson’s correlation.

Next, we apply the SVD - model based collaborative filtering. Here in table 8, we can see that it recommended top 5 names for user ‘A1B2C3D4’.

Table 8: Recommendation for ‘A1B2C3D4’ using SVD.

sl num	userId	nameId	Rating
0	A1B2C3D4	5193640278	5
1	A1B2C3D4	7946832105	5
2	A1B2C3D4	1357924680	5
3	A1B2C3D4	8192736450	5
4	A1B2C3D4	6512390847	5

Again, in table 9, recommendation of top 5 names for user ‘E5F6G7H8’ is shown.

Table 9: Recommendation table ‘E5F6G7H8’ using SVD.

sl num	userId	nameId	Rating
0	E5F6G7H8	5827369140	5
1	E5F6G7H8	4728901563	5
2	E5F6G7H8	5193640278	5
3	E5F6G7H8	5947683210	5
4	E5F6G7H8	8192736450	5

And, top 5 names for user ‘M3N4O5P6’ are generated and shown below in table 10.

Table 10: Recommendation table ‘M3N4O5P6’ using SVD.

sl num	userId	nameId	Rating
0	M3N4O5P6	1796253408	5
1	M3N4O5P6	8192736450	5
2	M3N4O5P6	7946832105	5
3	M3N4O5P6	3578204961	5
4	M3N4O5P6	6512390847	5

SVD model has a test rmse value of 0.0089 and cross validation rmse value of 0.0546. Compared to KNN With means, this model has a lower rmse value.

In terms of suggestions, each user is assigned a distinct set of names formed by filling in missing entries in the matrix during matrix factorization using SVD.

Table 11: Model Comparison-I.

Model	test_rmse	cv_rmse
SVD	0.0089	0.0546
KNN With Means	0.0727	0.0843

From analysing the results displayed in table 11, we can now clearly say that SVD is the better model comparing to KNN with means or popularity-based system with a superior rmse value of 0.0089.

Some more algorithms or models such as BaselineOnly, KNNWithZScore, CoClustering and NMF are applied on the dataset for a better understanding and the test-rmse, fit-time and test-time are generated and exhibited in table 12.

Table 12: Model Comparison-II.

Model	test_rmse	fit_time	test_time
BaselineOnly	0.967806	0.592147	0.375389
KNNWithZScore	1.044329	0.426561	1.312531
CoClustering	1.090479	9.427216	0.254634
NMF	1.132793	17.536839	0.325683

Based on the aforementioned results shown in both table 11 and 12, we can say that the BaselineOnly model performs well with the data because it has an rmse score of 0.968 and we can conclude that in all of the models used in this research, the SVD model is the best performing model with an rmse score of 0.0089.

6 CONCLUSION

Recommender systems recommend goods based on users' preferences. Popularity-based recommender models consistently recommend university names for all users. Collaborative filtering recommender systems, including KNN With means and SVD, have better rmse values and test-results. The SVD model is the best choice.

To conclude, this study aids Bangladeshi undergraduates in university selection using multiple user ratings and distributed machine learning for AI-driven education solutions.

7 FUTURE WORK

In future, we could work on other universities focused on different parts of the world.

REFERENCES

Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., and Wu, X. (2019). Towards federated learning at scale: System design. *arXiv preprint arXiv:1902.01046*.

Choi, J., Kim, J., and Cho, S. (2018). Distributed deep learning for large-scale recommender systems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2076–2084. ACM.

Ghoting, A., Talwalkar, A., and Dhillon, I. (2011). Distributed matrix factorization with mahout. In *Proceedings of the 2011 SIAM International Conference on Data Mining*, pages 111–122. Society for Industrial and Applied Mathematics.

Islam, M. A., Hossain, M. A., and Rahman, M. M. (2019). Real-time data stream clustering for recommender systems in big data. *IEEE Transactions on Big Data*, 5(1):130–143.

Jin, Z., Lu, K., and Liang, S. (2017). A hybrid distributed recommendation system combining collaborative filtering and content-based filtering. *Future Generation Computer Systems*, 75:98–108.

Kaur, H. and Chawla, M. (2020). A hybrid approach of collaborative filtering and deep learning for recommendation systems: A comprehensive review. In *2020 IEEE 7th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON)*, pages 1–6. IEEE.

Kumar, N. and Kaur, K. (2020). Scalability in distributed machine learning. In *Distributed Computing and Internet Technology*, pages 61–73. Springer.

Li, Y., Wang, S., Zhu, L., Zhang, Y., and Zhang, Z. (2014). A distributed collaborative filtering algorithm based on mapreduce for large-scale recommendation systems. In *2014 IEEE International Conference on Data Mining Workshops*, pages 513–520. IEEE.

Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., and Zaharia, M. (2016). Mllib: Machine learning in apache spark. *Journal of Machine Learning Research*, 17(34):1–7.

Wang, H., Xiong, J., Wang, C., Zhang, H., and Shi, Y. (2020). A distributed recommendation system based on lightfm in e-commerce. In *International Conference on Advanced Data Mining and Applications*, pages 119–130. Springer.

Yang, Q., Liu, Y., Chen, T., and Tong, Y. (2020). Federated collaborative filtering for privacy-preserving personalized recommendation system. *Future Generation Computer Systems*, 104:673–682.

Yao, Q., Liu, T., He, X., Wang, J., Gu, X., and Liu, T. (2019). Hybrid distributed recommendation system based on collaborative filtering and deep learning. *Applied Sciences*, 9(10):2017.

Yu, F., Seo, S., and Brinkhoff, T. (2020). Distributed machine learning for predictive personalization in intelligent transportation systems. In *2020 IEEE 17th Annual Consumer Communications & Networking Conference (CCNC)*, pages 1–6. IEEE.

Zhang, F., Yuan, N. J., Lian, D., Xie, X., and Ma, W. Y. (2019a). Distributed learning for large-scale heterogeneous collaborative filtering. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 1682–1691. ACM.

Zhang, Y., Liu, M., Zhang, Y., Tang, J., and Gao, H. (2019b). A distributed learning framework for personalized recommendation systems. *IEEE Transactions on Industrial Informatics*, 15(3):1471–1481.