Journal of information and
communication convergence engineering
http://jicce.org

J. Inf. Commun. Converg. Eng. 22(1): 7-13, Mar. 2024

Regular paper

U-Net-based Recommender Systems for Political Election System using Collaborative Filtering Algorithms

Nidhi Asthana[1] and Haewon Byeon[2]*

[1]Department of Applied Mathematics, STME NMIMS Indore campus, Indore, Madhya Pradesh 452005, India
[2]Department of AI-Big Data, College of AI Convergence, Inje University, Gimhae, 50834, Republic of Korea

Abstract

User preferences and ratings may be anticipated by recommendation systems, which are widely used in social networking, online shopping, healthcare, and even energy efficiency. Constructing trustworthy recommender systems for various applications, requires the analysis and mining of vast quantities of user data, including demographics. This study focuses on holding elections with vague voter and candidate preferences. Collaborative user ratings are used by filtering algorithms to provide suggestions. To avoid information overload, consumers are directed towards items that they are more likely to prefer based on the profile data used by recommender systems. Better interactions between governments, residents, and businesses may result from studies on recommender systems that facilitate the use of e-government services. To broaden people's access to the democratic process, the concept of "e-democracy" applies new media technologies. This study provides a framework for an electronic voting advisory system that uses machine learning.

Index Terms: Demographics, Collaborative Filtering algorithms, e-Government services, e-democracy

I. INTRODUCTION

Since their inception, recommender systems (RSs) have become an integral part of many types of online interactions, from tourism and social media to healthcare and smart cities [1-4]. RSs were originally developed to increase customer satisfaction and loyalty on e-commerce websites [5,6]. Massive volumes of data have been generated from the proliferation of Internet-connected devices (such as smartphones, tablets, laptops, and smart sensors) and sensors [7,8]. "Big data" [9] refers to datasets that are very large in bulk and are characterized by the "five V's": velocity, variety, veracity, volume, and value. The capacity to create and analyze large amounts of data has had a profound impact on many areas of daily life, such as communication on social media, online shopping, healthcare delivery, utility costs, and more [10]. Users may obtain valuable insights about their health, environment, and other topics through the correct processing of big data [11], which in turn helps them respond to changes in a timely manner. However, processing large datasets is difficult, notwithstanding the widespread nature of such applications.

Machine learning utilizes recommender systems to predict user likes and dislikes based on their past interactions with similar products. Recommender systems (also known as recommendation systems), are a subclass of information filtering systems that anticipate the "ating"or "reference"a user would accord to an item. Playlist makers for streaming video and music services as well as product developers for e-commerce sites are examples of the widespread use of recommender systems. The inputs for such systems may be simple, such as music, or complex, including numerous sources and

Received 19 August 2023, **Revised** 14 October 2023, **Accepted** 19 October 2024
***Corresponding Author** Haewon Byeon (E-mail: byeon@inje.ac.kr)
Department of AI-Big Data, College of AI Convergence, Inje University, Gimhae 50834, Republic of Korea

Open Access https://doi.org/10.56977/jicce.2024.22.1.7 print ISSN: 2234-8255 online ISSN: 2234-8883

media. Better product suggestions for clients or personalized recommendations among friends are two uses of recommender systems. Machine learning techniques are used by recommender systems to improve their anticipatory ability. A recommender system can use any number of available machine learning methods. The optimal algorithm for a given task may vary, depending on the characteristics of the data being processed. Currently, there is a major problem in finding relevant information online because of the exponential growth in online content. Increasing amounts of data are also becoming publicly accessible to political parties and candidates. This is becoming a major problem for voters in elections because they must choose representatives from a long list of candidates, especially when many candidates are unfamiliar to voters. The benefits of the work conducted in this area can be summarized as follows:

A connection is built between two fields of study with each gaining a lot from one another, as in machine learning and recommender systems (RSs). A one-of-a-kind roadmap is generated for researchers on machine learning-based RSs, covering obstacles, outstanding problems, and potential answers, to showcase the most important use cases. The state of the art is conveyed by comparing the capabilities of current frameworks for a range of criteria, such as calculation time and suggestion precision. A roadmap is laid out for the future to help overcome some of the problems plaguing machine learning-based RSs, to make them more effective.

The remaining sections of the paper are organized as follows. In Section 2, the primary difficulties of RSs are discussed, providing an overview. In Section 3, the suggested technique is presented along with a discussion of the most pressing outstanding questions. The experimental findings are presented in Section 4. In the final Section 5, a summary is presented emphasizing the significance of the suggested framework for RSs.

II. RELATED WORKS

Information-filtering systems such as recommender systems may alleviate information overload by highlighting relevant results from a large pool of data created in response to a user's stated or inferred interests, preferences, or past behavior with respect to a certain product or service. The difficulty of making sound judgements in the face of increasing amounts of data is known as the “information overload problem.” Recommender systems based on collaborative filtering (CF) overcame this issue [2] by predicting which of the numerous items users would enjoy. This came about as a result of the rapid rise in Internet usage during the early and mid-1990s. The purpose of a recommendation system is to leverage user information to offer useful product or service suggestions [3]. While consumers have more options than ever before, online retailers and service providers find it harder than ever to reach customers through targeted product ads. User preferences for media, reading materials, vacation spots, and online resources are compiled by the recommender system. Word-of-mouth is translated into digital form with the help of CF for an algorithm that suggests products favored by people with similar preferences. Although CF has been widely used as a technique for making recommendations, it has been plagued by issues such as performance deterioration due to data sparsity, cold-start, long-tail issues, and scalability concerns [4]. Improvements in recommender system speed can now be achieved using a number of different methods owing to the structural improvements made possible by deep learning. Because deep-learning-based recommender systems can solve the issues of traditional CF models (such as data sparsity, cold start, scalability, and long tail) and produce high-quality recommendations, their development has garnered considerable attention [5]. These systems use restricted Boltzmann machines to extract latent features of user preferences or ratings, and an undirected two-layer graphic model as a graphical probabilistic model to make predictions. A deep belief network, which is a multilayered neural network, is used to extract high-level data from low-level information for user preferences. The autoencoder is a deep-learning model that reduces the dimensionality of the user item matrix to extract additional latent characteristics from the encoder output. Accurate suggestions are provided by tracking changes in user behavior over time using a recurrent neural network (RNN), which is a deep-learning model optimized for handling sequence data. In addition, the convolutional neural network (CNN), which excels in tasks such as image recognition and object classification, provide strong performance for recommender systems [6], by extracting latent variables and features from raw data such as voice, text, and image data. A recommender system helps viewers who are having trouble locating the anchors of interest. To record the differences in taste between anchors and viewers, multihead-linked device is created in this study, to extract relevant functions from an expression. The findings show that the proposed model is much more effective than current top-tier recommendation algorithms. To forecast a user's preference or propose products suited to preferences, deep learning-based recommender systems have been created [7]. These systems can infer the hidden meaning between the user and the item from different types of input data, such as photographs and unstructured data.

III. OVERVIEW

Computer-based approaches known as recommender systems have been developed to help users choose items they are more likely to prefer. This method is most commonly used in online retail, where it is used to recommend products that a shopper is more likely to prefer. According to Yager (2003), a recommender system is a "participatory" system in which the user voluntarily submits information about his preferences, in contrast to targeted marketing, which collects this data passively. Extensional information, which is simply knowledge based on actions or prior experiences regarding certain items, is the basis of the suggestions in targeted marketing. Users and products are the two primary components of an online store's recommender system. The point of using a recommender system in an online store is to boost product sales. The following are some of the most common issues with recommender systems, as cited by [8]. False positives (presenting goods that consumers do not like) should be minimized to ensure that the recommender system information is trustworthy. The success of a recommendation system can be gauged by the number of consumer suggestions. Because recommender systems rely on commonalities between users, the sparsity issue of recommender systems arises when the number of rated items is small compared to the total number of items. The computational cost of recommender systems increases with the number of users and goods. It is impossible to convey user correlations unless the user has purchased similar items with similar ratings. The recommender systems are unable to correlate goods that have distinct names but are otherwise in the same category. To be recommended, a product must have received at least one rating from an existing client. Users may have difficulty in articulating their feelings about a product. Consequently, contradictory advice is often provided.

IV. METHODOLOGY

State-of-the-art recommendation engines rely heavily on the synergy between several forms of machine learning technology and data. Google identifies four main processes that occur within a recommender system. Information gathering, storage, analysis, and filtering fall under this category. Let us break down each stage and examine how it works. Data collection is the first step in developing a user profile for the prediction model. Details about the user's qualities, habits, or information found in the resources that a person has access to, might be included in the data. Recommendation engines mostly use two distinct types of data: explicit or user input data (such as star ratings, likes/dislikes, reviews, and product comments) and implicit or behavior data (such as viewing an item, adding it to a wish list, and the amount of time spent on an article). Although analyzing implicit data is more challenging, it is simpler to gather, not requiring any action on the part of the consumer. Users may not always be

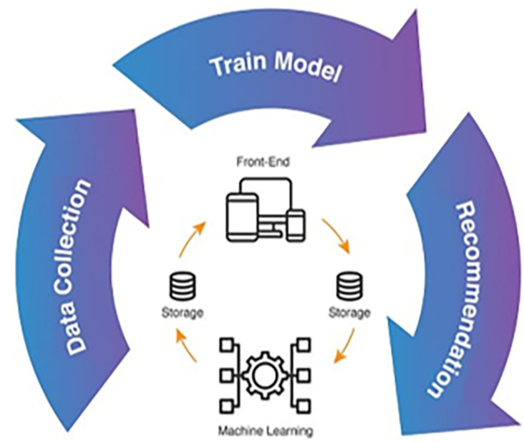


Fig. 1. Data storing

willing to undertake extra efforts to provide extensive explicit data although such information would be more reliable. Recommender systems may utilize user attribute data (such as age, gender, country, and hobbies) and item attribute data (genre, kind, and category). In our discussion of data preparation for machine learning, we stress the need for sufficient information to properly train a model. The more useful the information provided by the algorithms, the better their predictions will be. The next step is to choose a storage medium that can grow as needed to accommodate the accumulated information. The storage method should be tailored to the specifics of the data to be used to make suggestions. Structured data may be stored in a traditional SQL database, whereas unstructured data can be stored in a NoSQL database; both types of data can be stored in a cloud data warehouse, or a data lake can be used for Big Data initiatives. Alternatively, several data stores may be used in concert with each other.

Data analysis is essential for effective use. Depending on how rapidly the system provides suggestions, several forms of data analysis can be used. Data are handled and analyzed in batches on a periodic basis. A good example is daily sales figures. Data being processed and analyzed every few minutes or seconds instead of in real time, is called near-real-time analysis. These suggestions can be made within the course of a single web-surfing session. When data are analyzed in real-time, they are generated as they are being processed. Therefore, the algorithm generates suggestions for both the here and now. Selecting the best filtering strategy and establishing the most effective algorithm to train a model are crucial steps in developing a recommender system. Distance-measuring algorithms are examples of relatively simple algorithms although they may also be sophisticated and resource intensive.

A. UNET — Network Architecture

In 2015, Ronneberger et al. created UNET, an architecture for biomedical image segmentation, at the University of Freiburg, Germany. Currently, UNET is one of the most frequently used strategies in semantic segmentation tasks. This neural network is completely convolutional and optimized for learning using fewer training examples. The original fully convolutional networks (FCN) for semantic segmentation were created by Long et al. in 2014; however this new method is superior. UNET is an encoder-decoder network design in the form of a U with a bridge connecting four encoder blocks and four decoder blocks. In the encoder network (contracting route), half the spatial dimensions and twice the number of filters (feature channels) are used at each encoder node. Similarly, the decoder network halves the number of feature channels, while doubling the spatial dimensions. Using a series of encoder blocks, the encoder network extracts features from the input image and learns the abstract representations. Two 3×3 convolutions, each followed by a rectified linear unit (ReLU) activation function, constitute each encoder block. Nonlinearity is introduced into the network through the ReLU activation function, allowing for an improved generalization of the training data. The ReLU output is connected to the matching decoder block's skip input. Subsequently, the spatial dimensions of the feature maps (height and width) are halved using a 2×2 max-pooling. Because there are fewer trainable parameters, less computational power is required. These skip connections provide more data to the decoder, thereby improving the quality of the semantic characteristics produced. Indirect gradients flow more easily to the lower levels through this shortcut link. In other words, the network can learn a more accurate representation when skip connections are used during backpropagation. Information flow is completed by a bridge that links the encoder and decoder networks. Each convolution is followed by a ReLU activation function in all four layers. Using the abstract representation, the decoder network creates a semantic segmentation mask. In the decoder, a 2×2 transpose convolution is performed as the first step. The next step is to join it with the appropriate skip connection feature map of the encoder block. In cases where features from lower levels are lost because of the network depth, these skip connections are restored. Subsequently, a pair of 3×3 convolutions is utilized, along with a ReLU activation function. A sigmoid-activated 1×1 convolution is applied to the output of the final decoder. The segmentation mask representing the pixel-wise categorization is provided by the sigmoid activation function.

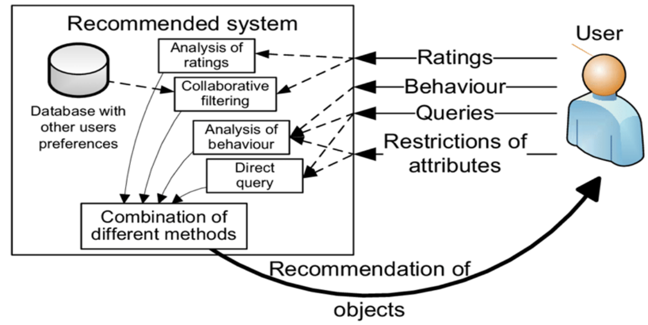


Fig. 2. Data filtering

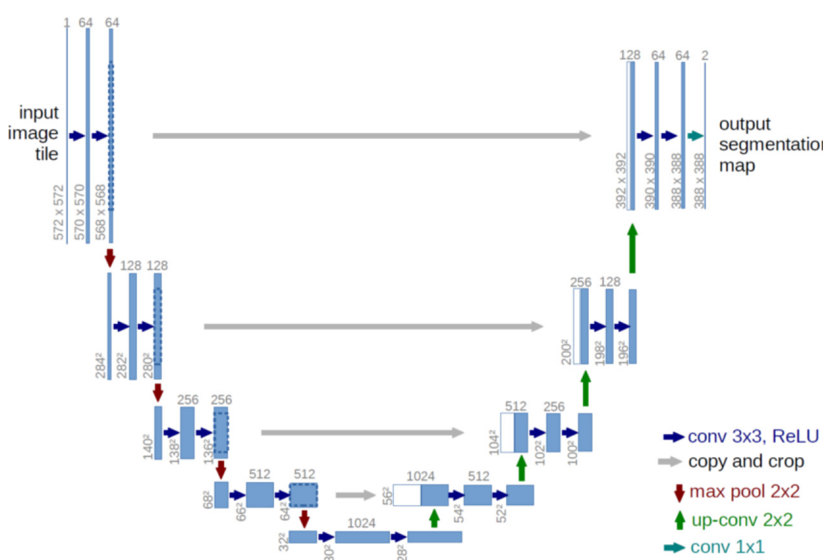


Fig. 3. Skip connections

B. Hybrid Models and Deep Learning

Our state-of-the-art recommendation engine algorithms integrate collaborative filtering with content-based models using deep learning. More nuanced interactions between users and goods can be learned using hybrid deep-learning algorithms, and they are less likely to oversimplify user preferences because of their nonlinear nature. Even from cross-domain datasets (such as those that include both music and films or TV episodes), deep learning models can accurately discern complicated preferences across a wide variety of products.

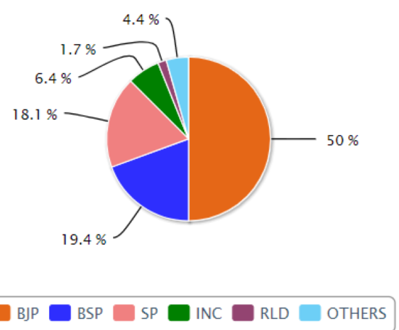


Fig. 4. Election Results

Table 1. 2019 Election results Uttar Pradesh

Party	Seats	Votes %
BJP	62	50
BSP	10	19.4
SP	5	18.1
ADS	2	1.2
INC	1	6.4

Table 2. 2022 Vidhan Sabha / Assembly election results Uttar Pradesh

Party	Seats	Votes %
BJP	255	41.6
SP	111	32.3
ADS	12	1.6
RLD	8	2.9
NINSHAD	6	0.9
Others	11	20.7
NOTA	6,37,304 (0.7%)	

Electors: 15,19,44,079
Votes Polled: 9,15,25,592
Turnout: 60.2%
Total ACs: 403
General: 318 | SC:85

Despite their widespread use, measures to gauge the quality of a suggestion are lacking in important respects. Determining the actual quality of a suggestion requires coupling it with user data. A better picture of the quality of the suggestions may be obtained by monitoring metrics such as recommendation hit rates, platform engagement, and user responses.

Table 3. 2022 Vidhan Sabha election summary of Varanasi Cantonment 2022

Candidate Name	Party	Votes	Votes %	Margin
Saurabh Srivastava	BJP	1,47,833	61.0	-
Pooja Yadav	SP	60,989	25.2	86,844 (35.8 %)
Rajesh Kumar Mishra	INC	23,807	9.8	1,24,026 (51.2 %)
Kaoshik Kumar Pandey	BSP	7,068	2.9	1,40,765 (58.1 %)
Rakesh Pandey	AAP	1,165	0.5	1,46,668 (60.5 %)

Table 4. 2022 Vidhan Sabha election summary of VARANASI NORTH

Candidate Name	Party	Votes	Votes %	Margin
Ravindra Jaiswal	BJP	1,34,471	55.0	-
Ashfaque	SP	93,695	38.3	40,776 (16.7 %)
Shyam Prakash	BSP	10,457	4.3	1,24,014 (50.7 %)
Gulerana Sabassum	INC	3,102	1.3	1,31,369 (53.7 %)
Harish Mishra	AIMIM	1,643	0.7	1,32,828 (54.3 %)

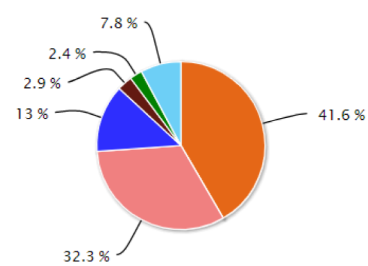


Fig. 5. Election results

Retraining recommenders depending on fresh ratings or interactions from users is also important, to update suggestions when a user has not engaged for a certain length of time. The way the recommender handles novelty, diversity, and selection bias should also be considered and whether the user is restricted to a subset of items. These key performance indicators (KPIs) are often monitored using A/B testing. The hybrid filtering technique is a type of customization that takes advantage of the synergistic benefits acquired from using a variety of recommendation approaches. The algorithm that Netflix uses to propose shows and films is a good example of a hybrid system. Both content-based and collaborative filtering are used to make film recommendations for visitors of the site.

V. EXPERIMENTAL RESULTS

Root mean-square error (RMSE), mean absolute error (MAE), and k-fold cross-validation are examples of common statistical accuracy metrics that can be used to assess recommender accuracy. Comparable to MAE, a harsher punishment can be used for erroneous predictions and a more lenient punishment for those that are quite close to the mark, such as squaring the difference between the actual and anticipated numbers rather than adding them. If the difference is greater than zero, the final value is larger, and vice versa, for smaller differences. The better the results, the smaller is the RMSD.

With access to a product's metadata suggesting similar products to the consumer becomes simpler. Netflix's vast meta-data tags and a distance metric can be used to propose other films to watch. A second option is to employ natural language processing methods such as term frequency-inverse document frequency (TF-IDF) to express movie descriptions as vectors. All that remains is deciding on the measure of similarity. The Jaccard similarity, Euclidean distance, and Pearson's coefficient are most often used. In a user-focused system, the lengths of the lines between users represent the gaps between them. The available products comprise the preference area, whereas the user ratings serve as the axes. We look for products that people with similar likes enjoy. The closer two people are, the more likely they are to have the same taste.

$$d(x,y)=\sqrt{(x_1-x_2)^2+(y_1-y_2)^2} \tag{1}$$

The accuracy with which recommender system ratings correspond to user ratings is a topic of predictive measurement. They can be used for jobs that do not have a clear yes or no answer. The most widely used and easily understood prediction metrics are the MAE and RMSE. In practice, when evaluating collaborative recommendation models on a K-fold cross-validated dataset, both the RMSE and MAE are often examined. Non-accuracy measures used for scoring are detailed below; however, from a business viewpoint, what counts is not only the greatest RMSE or MAE. In the future, metrics for binarized recommendation jobs will be discussed.

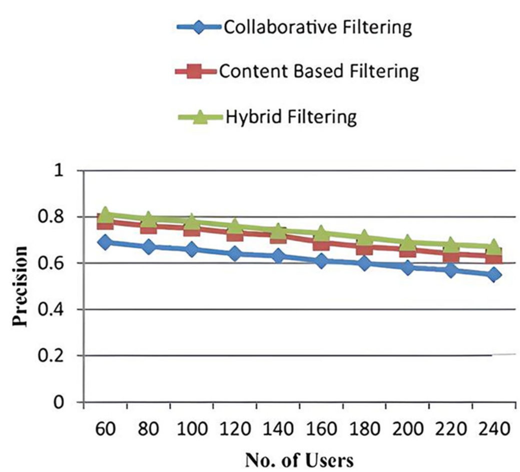


Fig. 6. Precision for recommender system

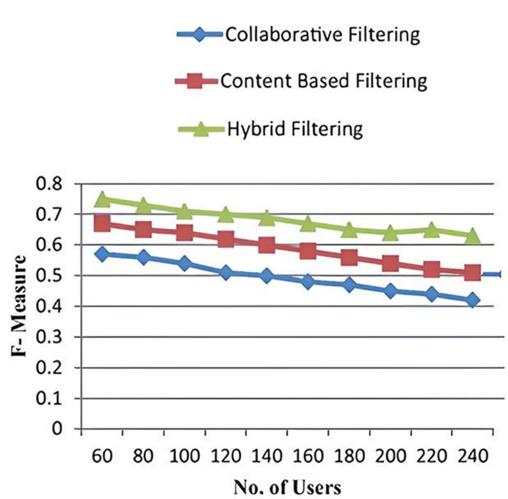


Fig. 7. F-Measure of recommender system

VI. CONCLUSION

In this study, we presented a hybrid personalized recommender system using the TensorFlow framework and UNet. Accurate recommendations can be made using an election-based system that records candidates' past actions over time. The proposed UNet-based recommender systems predict voter preferences for candidates in real time. The new candidate suggestion performance varies by data type, but the prediction performance is generally good. However, considering all available past data, the suggested approach cannot guarantee consistent advisory performance. In other words, the method performs better when applied to large amounts of historical data. The larger the dataset, the more stable is the training and the more accurate is the suggestion when utilizing U-Net, as opposed to the conventional recommendation method. U-Net is an improvement over previous methods that overcomes their flaws and boosts the overall performance. Experimental findings on a real dataset demonstrate the effectiveness of the proposed U-Net in reducing the burden of processing recommendations and significantly increasing speed and accuracy. To create a time-aware individualized recommendation system, precise time-division models and alternative deep learning models will be further investigated.

ACKNOWLEDGEMENTS

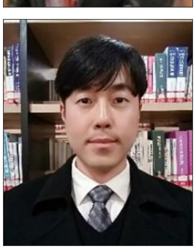
This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education (NRF- RS-2023-00237287, NRF-2021S1A5A8062526) and local government-university cooperation-based regional innovation projects (2021RIS-003).

REFERENCES

[1] J. L. Hennessy and D. A. Patterson, "Instruction-level parallelism and its exploitation," in *Computer Architecture: A Quantitative Approach,* 4th ed., San Francisco, CA: Morgan Kaufmann Pub., pp. 66-153, 2007.
[2] Y. Deldjoo, M. Schedl, P. Cremonesi, and G. Pasi, "Recommender systems leveraging multimedia content," *ACM Computer Survey*, vol. 53, no. 5, pp. 1-38, Sep. 2020. DOI: 10.1145/3407190.
[3] M. Hong and J. J. Jung, "Multi-criteria tensor model for tourism recommender systems," *Expert Systems with Applications*, vol. 170, pp. 114537, May 2021. DOI: 10.1016/j.eswa.2020.114537.
[4] S. M. Ghafari, A. Beheshti, A. Joshi, C. Paris, A. Mahmood, S. Yakhchi, and M. A. Orgun, "A survey on trust prediction in online social networks," *IEEE Access,* vol. 8, pp. 144292-144309, 2020. DOI: 10.1109/ACCESS.2020.3009445.
[5] J. Saha, C. Chowdhury, and S. Biswas, "Review of machine learning and deep learning-based recommender systems for health informatics," *Deep Learning Techniques for Biomedical and Health Informatics,* vol. 78, pp. 101-126, 2020. DOI: 10.1007/978-3-030-33966-1_6.
[6] L. Quijano-Sanchez, I. Cantador, M. E. Cortés-Cediel, and O. Gil, "Recommender systems for smart cities," *Information Systems,* vol. 92, pp. 101545, Sep. 2020. DOI: 10.1016/j.is.2020.101545.
[7] R. Katarya and N. Verma, "Automatically detection and recommendation in collaborative groups," in *Proceedings of the International Conference on Intelligent Sustainable Systems*, Palladam, India, pp. 218-222, 2017. DOI: 10.1109/ISS1.2017.8389400.
[8] B. Deebak and F. Al-Turjman, "A novel community-based trust aware recommender systems for big data cloud service networks," *Sustainable Cities and Society,* vol. 61, pp. 102274, Oct. 2020. DOI: 10.1016/j.scs.2020.102274.
[9] G. Gupta and R. Katarya, "A study of recommender systems using Markov decision process," in *Proceedings of the Second International Conference on Intelligent Computing and Control Systems*, Madurai, India, pp. 1279-1283, 2018. DOI: 10.1109/ICCONS.2018.8663161.
[10] R. Katarya and O. P. Verma, "Recommender system with grey wolf optimizer and FCM," *Neural Computing and Applications*, vol. 30, no. 5, pp. 1679-1687, Dec. 2016. DOI: 10.1007/s00521-016-2817-3.
[11] M. Fu, H. Qu, Z. Yi, L. Lu, and Y. Liu, "A novel deep learning-based collaborative filtering model for recommendation system," *IEEE Transactions on Cybernetics,* vol. 49, no. 3, pp. 1084-1096, Mar. 2018. DOI: 10.1109/TCYB.2018.2795041.

Nidhi Asthana

Dr. Nidhi Asthana is an Assistant Professor at the School of Technology Management & Engineering (STME), NMIMS, Indore. She has experience working with engineering institutes in Indore and teaches courses in Mathematics and Business Analytics.



Haewon Byeon

Dr. Haewon Byeon earned his DrSC degree from Ajou University Medical School in South Korea. He is an Assistant Professor in the Department of AI-Big Data at Inje University. Additionally, he is the director of the INJEM (INJE university Medical big data) lab, an open laboratory certified by the Ministry of Science and ICT in South Korea.