ISSN: 2584-0495 Vol. 2, Issue.3, pp. 678-684

International Journal of Microsystems and IoT

ISSN: (Online) Journal homepage: https://www.ijmit.org

Mentor Recommendation System Using KNN Item-Based Collaborative Filtering

Mehak Agrawal, Ashir Vijh, Anand Kumar Dubey, Anuranjana

Cite as: Agrawal, M., Vijh, A., Dubey, A. K., & Anuranjana. (2024). Mentor Recommendation System Using KNN Item-Based Collaborative Filtering. International Journal of Microsystems and IoT, 2(3), 678-684. https://doi.org/10.5281/zenodo.11125944

Published online: 11 March 2024

DOI: https://doi.org/10.5281/zenodo.11125944

International Journal of Microsystems and IoT
Vol. 2, Issue.3, pp. 678-684; DOI: https://doi.org/10.5281/zenodo.11125944

Check for updates

Mentor Recommendation System Using KNN Item-based Collaborative Filtering

Mehak Agrawal, Ashir Vijh, Anand Kumar Dubey, Anuranjan

Department of Information Technology, Amity University Uttar Pradesh, India

ABSTRACT
Mentors play a critical part in a person's success. In today's world, many people, especially students in universities, are struggling to search a mentor and even if they find one, they don't find them matching their goals and personality. This creates a problem of lack of proper guidance that people need to move further not only in career but in life as well. In most existing online mentorship systems in universities, matching mentors and mentees is a manual job. The proposed solution is to build a mentor recommendation system in which the user is connected with the appropriate mentor based on the mentee's needs. With this mentor referral system, the battle to find the perfect mentor will be over. This recommendation system uses KNN item based collaborative filtering technique instead of support vector machine method to make it more efficient. An online survey has been performed to find the difficulties faced by the students with respect to the mentors allotted to them. The usage of recommender systems, as well as trust and reputation processes, can aid in the elimination of manual matchmaking and allow for a more workable and active perspective that adjusts according to the needs of the users. Having the correct pair instead of randomly matching students would be more advantageous, not only in mentoring, but also in promoting the student in his field of interest, because the mentor can guide the student or mentee in expanding his or her horizons and go deeper in that subject and passion.

KEYWORDS
collaborative filtering, k-means algorithm, mentor – mentee, matching system, recommendation system

1. INTRODUCTION

The majority of the internet services we use today are powered by recommender systems. Youtube, Netflix, Amazon, Pinterest, and a slew of other internet services rely on recommender systems to sift through millions of pieces of content and make personalised recommendations to their users. Mentors and mentees can be matched using recommender systems, and trust & reputation mechanisms can be used to enhance the decision-making process [1]. Searching the right mentor and nurturing a relationship with them can be difficult, and virtual intimacy can be difficult to achieve, especially if two people have never met in person [2]. Because the mentor and the student can be the same person at the same time, identifying the mentor and the student in the online community can be difficult [3]. Two-Sided Matching is a thorough method for determining matchings & allocations based on the preferences of the participants [4]. The recommender system has come out as a major research interest, with the goal of assisting users in finding things online by providing recommendations that closely match their preferences [5]. Personalized recommendations attempt to understand the user's characteristics and preferences by collecting and analysing historical behaviour to determine what kind of person the user is, what kind of behaviour preferences the user has, what kind of things the user likes to share, and so on, and finally understand that user's characteristics and preferences based on the platform's rules and suggest the products and information which the user is looking for. Personal interest can cause the recommendation system to recommend items to meet users' individualities, particularly for experienced users [6]. The mentor recommendation system works as a filtration system that aims to forecast a student's preference for a domain-specific item. This domain-specific component is 'mentor' in this study; hence the main focus is on filtering and recommending only those mentors who students would like based on their feedback. The survey conducted shows the problems faced by students in universities because The research shows the comparison why KNN Item Based Collaborative Filtering was used instead of other filtering techniques like support vector machine, content based and user based collaborative filtering.

2. SURVEY OF PROBLEMS FACED BY STUDENTS FROM ALLOCATION OF MENTORS

266 university students participated in the survey conducted to find the problems faced by students with respect to the mentors allotted to them by the university.

Table 1. Survey Research Questionnaire

S. No.	Problem	Yes	No
1.	Is the mentor allotted to you possesses the domain knowledge of your project area?		
2.	Are you satisfied with the guidance provided by your mentor?		

3.	Is the mentor allotted to you able to clear your project related doubts?
4.	Does your allotted mentor give you new ideas or valuable suggestions?

2.2 Survey Research Results

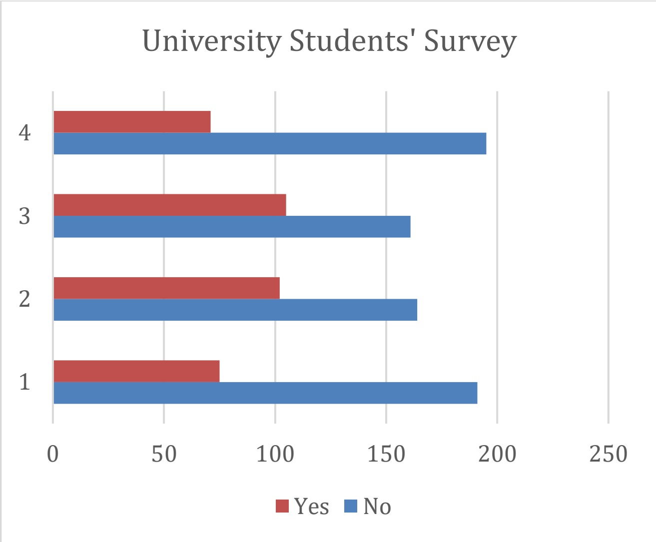

Fig. 1 Plot showing the result of survey research

The survey results showed that out of 266 students, 71.8% mentors allotted to students do not possess the domain knowledge of the projects that their mentors have undertaken. 61.65% students were not satisfied by the guidance provided their mentors due to lack of compatibility between mentors and mentees. 60.5% mentors were not able to clear all the doubts of mentees because of the lack of domain knowledge. 73.3% mentees were not able to get valuable suggestions and new ideas from their mentors. This study showed the problems faced by the students in university because of the random allotment of mentors and justified the need of having a mentor-mentee matching approach in the universities.

3. METHODOLOGIES

3.1 Filtering Approaches

3.1.1 Content-based filtering

A content-based filtering system chooses items based on the correlation between the items' content and the user's preferences, as opposed to a collaborative filtering system, which chooses items based on the correlation of people with similar preferences [7]. This method of filtration is developed on the mentor data provided. The system suggests mentors who are comparable to those who have previously piqued a student's interest. This resemblance is based on information about the mentors or faculties and the student's previous preferences.

3.1.2 Collaborative filtering

This filtration approach is developed on a comparison and contrast of the student's behaviour with the behaviour of other different student's behaviour recorded in the database. The algorithm is deliberately reliant on all students' past performances. Traditional collaborative filtering approaches suffer from the cold-start problem: when a new item enters the system, it cannot be recommended until a sufficient number of people rate it [8]. Collaborative filtering is a method of making automatic predictions about students' interests by collecting data from a huge number of other students [9]. It does not rely on the data of only one student for modelling.

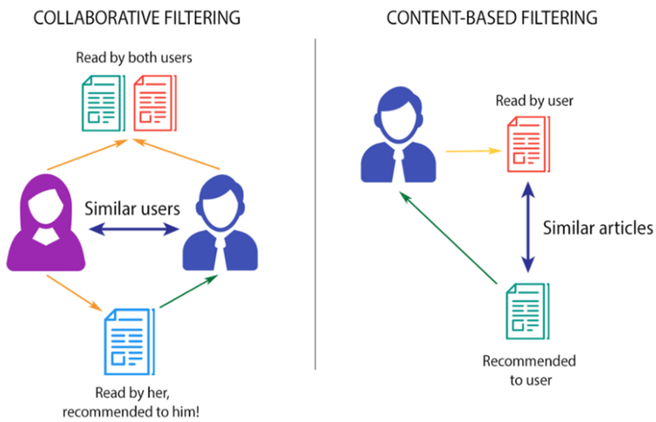

Fig. 2 Collaborative v/s Content-based filtering illustration

Collaborative filtering algorithms are categorized into two types:

3.1.2.1 User-Based Collaborative Filtering

In this filtering approach, the job is to predict the utility of items to a specific user based on a database of user votes from a sample or population of other users [10]. The objective here is to discover students who have similar historical preference patterns as student 'A,' and then offer mentors who are loved by those comparable students but whom 'A' hasn't yet met. This is accomplished by constructing a matrix of items that each user has viewed/clicked/liked/rated based on the job at hand, computing the similarity score between the students, and then recommending mentors that the concerned student is unaware of but that students similar to him/her are aware of and liked.

3.1.2.2 Item-Based Collaborative Filtering

Techniques based on items first examine the user-item matrix to identify relationships between different items, and then use these relationships to indirectly calculate recommendations for users [11]. The goal here is to find similar mentors and then recommend them to 'A' based on the student's past choices. It is achieved by identifying every pair of mentors that has been viewed/clicked/liked/rated by the same student, calculating the similarity of those rated/viewed/liked/clicked across all users who rated/viewed/liked/clicked both, and then recommending them according to their similarity scores. Item-based filtering is stable and faster than user-based filtering in a system with more users than items. When the rating matrix is sparse, it has also been demonstrated to outperform the user-based approach.

3.2 Classification Method: K-Nearest Neighbours Algorithm

We propose to use KNN algorithm along with item-based collaborative filtering (CF) for building the mentor recommendation system. The item-based KNN CF algorithm is a memory-based neighbour-based approach. Instead of calculating user similarity, the algorithm calculates the similarity between items rated (purchased or observed) by the target user and item [12]. KNN is an excellent go-to model for implementing item-based collaborative filtering, and also a solid foundation for developing recommender systems. But what exactly is the KNN? KNN is a lazy, non-parametric learning method. It makes inferences for new samples using a database and there, the data-points are classified into clusters. KNN makes no assumptions about the distribution of the underlying data and instead relies on item feature similarity. The K-nearest neighbours (KNN) algorithm recommends using similarity matrices; however, several disadvantages associated with the traditional KNN algorithm have been identified [13]. When KNN makes a prediction about a mentor, it calculates the "distance" between the target mentor and every other mento in its database, ranks the distances, and returns the top K nearest neighbour mentors as the most similar mentor recommendations. But how do we feed the ratings data frame into a KNN model? First, we must convert the ratings data frame into an arrangement that can be consumed by a KNN model. The data should be stored in a m x n array, where ‘m’ is the no. of mentors and ‘n’ is the no. of users. To reshape the ratings data frame, we'll pivot it to a wide format with mentors as rows and users as columns. Then, because we'll be performing linear algebra operations, we'll fill in the missing observations with 0s (calculating distances between vectors). This new data frame can be called as a "data frame of mentor features." The mentor feature data frame is an extremely sparse-matrix with the dimensions 13,500 x 113,291. Following the calculation of user similarity, the algorithm selects a number of students with the highest similarity as the U's neighbour, denoted as u'. After setting a fixed value K for neighbour selection, select only the users with the highest K similarity as neighbours, regardless of their neighbour similarity [Cui, B. B. (2017)]. To select the K that is best for your data, we run the KNN algorithm numerous times with different K values and select the K that reduces the no. of errors while retaining the algorithm's ability to make correct predictions when given data it has never seen before. When we reduce the K-value to one, the predictions become less precise. As we increase K's value, the predictions become more reliable due to the majority vote/average, and thus most likely to be accurate (up to a certain point). We eventually begin to notice a rise in the no. of errors. When we take a majority vote among labels (for example, selecting the mode in a classification problem), we usually make K an odd number to have a tiebreaker.

3.3 Similarity Matrix: Cosine Similarity

Cosine similarity is used to quantify the similarity of two things of any type [14]. It compares two documents on a scale that has been normalised [15]. The two samples can be strained from the same or different distributions. The no. of attributes in the two samples should be the same.

$$similarity(A,B) = \frac{A.B}{||A||X \vee |B| \vee = \frac{\sum_{i=1}^{n} A_i X B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} X \sqrt{\sum_{i=1}^{n} B_i^2}}} \quad (1)$$

The cosine similarity ranges from -1 to 1.1 and that means two random variables are perfectly positively correlated, -1 means that 2 random variables are perfectly negatively correlated, 0 means that 2 random variables are not correlated.

4. RESULT AND FINDING

The current mentoring system in many universities is a manual job and mentors are allotted to the students randomly. This issue is addressed by the suggested mentor recommendation system. This system can eradicate the manual match making of mentors and mentees. This can prove to be a more flexible approach towards mentorship system in universities where students will be recommended mentors based on their interests. The collaborative and KNN are not used together in existing systems. As a result, the proposed solution performs much more consistently. The existing system employs content-based filtering, which is unreliable and inconsistent in its performance over time with larger datasets. The proposed solution employs a collaborative filtering strategy that combines the usage of both, the KNN algorithm and cosine similarity to create a more unique solution. Numerous studies have led to the conclusion that collaborative-based when compared to content-based filtering, the recommendation performs significantly better. Even when there is a significant amount of data used, it is more convenient and requires fewer computation resources. This mentor recommendation system uses item-based collaborative filtering because the problem with the user-based formulation of collaborative filtering approach is its lack of scalability: for the aim of making predictions, it requires a real-time comparison of the target-user to all user records. Item-based CF is a form of this approach that addresses this issue. The proposed mentor recommendation system recommends the best mentor using the ratings given by the students (users) to the mentors. The figures below show that which mentors are rated by many students and which mentors have got the best ratings.

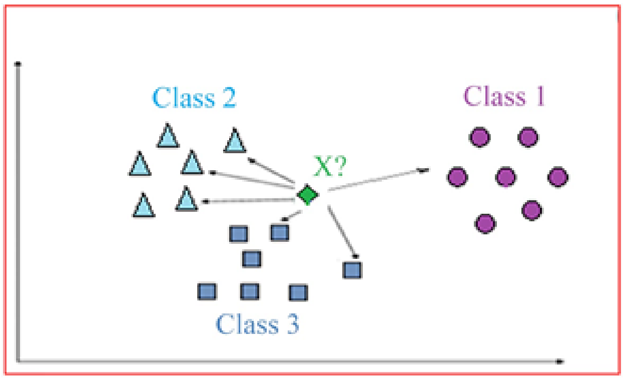

Fig. 3 Overview of KNN algorithm

K-Nearest Neighbour (KNN) was preferred in lieu of Support Vector Machine (SVM) due to the following reasons [21-23]:

Table. 2 Comparative study of KNN and SVM

	K-Nearest Neighbour (KNN)	Support Vector Machine (SVM)
1	Attempts to approximate the underlying distribution of the data in a non-parametric fashion.	Assumes there exist a hyper-plane separating the data points (quite a restrictive assumption).
2	KNN is simple vis-a-vis SVM and easier to implement too. The algorithm requires minimal tuning parameters and does not require large training datasets due to its lazy learning strategy, easily adapts to new data which makes it suitable for our use case.	More complex to implement, requires more extensive parameter tuning and larger datasets. Therefore, re-training the model would be computationally expensive.
3	Suitable for use cases where the data is updated continuously or erratically.	Relies on assumptions about data sets, such as linearity or normality, which can sometimes result in false results, relies on kernel methods, which take more time to compute.
4	Makes less assumptions about the distribution of the underlying data as well as works more efficiently with heterogeneous distributed points, which means that it is often efficient even when the distributional assumptions are not complete.	SVMs base their decisions on assumptions about some kind of distributional structure between dataset features, which may limit its effectiveness with complex patterns.

The plot of number of users voted for mentors shows the average no. of votes received by the mentors and shows that for which mentors the votes from the students or users are the highest. The mentors in the plot are represented by their Mentor IDs.

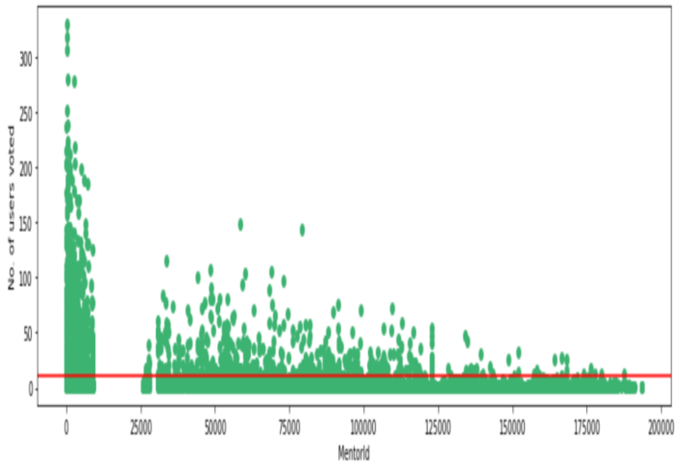

Fig. 4 Plot of number of users voted for mentors

The plot of number of votes by every user or student shows the average no. of votes given by the students to the mentors.

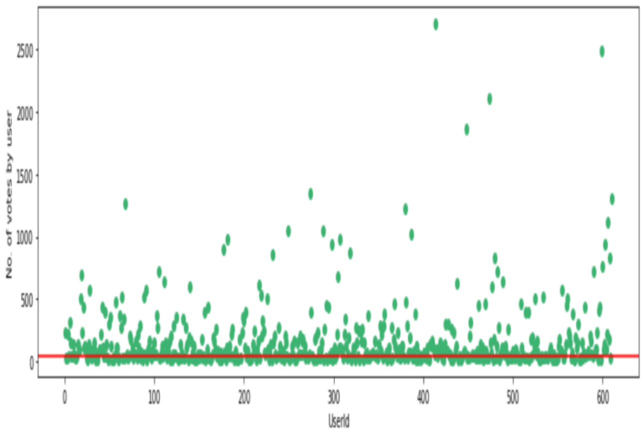

Fig. 5 Plot of number of votes by every user

Table. 3 Comparative study of existing publications

S. No.	Title	Advantages	Disadvantages
1.	Personalized Mentor/Mentee Recommendation Algorithms for Matching in e-Mentoring Systems [16].	The proposed algorithm matches on the basis of personal preferences of secondary students.	The algorithm doesn't consider the qualification of the mentors.
2.	Context-Based Hybrid Semantic Matching Framework for E-mentoring System [17]	The proposed framework uses the preferences of both mentors and mentees.	Some irrelevant attributes like age, gender, religion is used for making the matching framework.

3.	Two-Sided Matching for mentor-mentee allocations — Algorithms and manipulation strategies [18]	This study shows that multi-objective heuristics have the added benefit of producing solutions with higher quality across multiple criteria.	This doesn't consider the ratings of the mentors.
4.	Mentor-spotting: recommending expert mentors to mentees for live trouble-shooting in Codementor [19]	This system recommends mentors based on the availability, proximity, activity, and capability of the mentors.	This system does not consider the preferences of the students.

Table 2 shows the comparative study of existing papers and shows the advantages and disadvantages of existing solutions.

Table. 4 Comparison between collaborative filtering and content-based filtering

Collaborative Filtering	**Content Based Filtering**
System can be expanded.	System cannot be expanded as the student does not choose different type of mentors.
Mentors can get a lot of exposure.	Mentors do not get much exposure to the student.
The interaction of all students with the mentors influences the recommendation algorithm.	Only the concerned student's data is taken into account.
Higher and more consistent performance.	Unreliable and inconsistent in its performance over time with larger datasets.
Convenient and requires fewer computation resources.	Less convenient and requires more computation resources.

Table 3. shows the comparison between two most commonly used filtering approaches and shows that collaborative filtering is better in terms of scalability and performance.

Table. 5 Comparison between user-based collaborative filtering and item-based collaborative filtering

User-Based Collaborative Filtering	**Item-Based Collaborative Filtering**
Students' preferences change from time to time, and because this algorithm is based on student similarity, it may detect initial similarity patterns between two students who, after a while, may have completely different preferences.	Unlike students' preference, mentors don't change.
Because there are many more students than mentors, it becomes difficult to maintain such large matrices, which must be recomputed on a regular basis.	
This algorithm is extremely vulnerable to shilling attacks, in which fake student profiles with biased preference patterns are used to manipulate key decisions.	Shilling attacks are much harder because mentors cannot be faked.
This filtering is slower and less stable because of more users than items.	This filtering is faster and more stable.
It does not work well because the average rating given by a user to different items changes quickly.	It works well because the average rating given to an item usually does not change as quickly.

Table 4. shows the comparison between two types of collaborative filtering approaches and shows that item-based collaborative filtering has more advantages over user-based collaborative filtering.

5. CONCLUSION

The process of matching mentors to their mentees is critical to achieving a high standard in mentoring relationships [17]. The survey conducted showed the problems faced by students in universities because of the random allotment of mentors and justified the need of proposed mentor recommendation system for universities that uses the KNN item based collaborative filtering algorithm that is faster, addresses the problem of scalability, and also the shilling attacks are harder in this approach. The proposed system can also help to eradicate the manual match making job in the universities and help the universities to save time. This dynamic approach towards online mentorship in universities not only help the universities but also help in promoting the growth of students.

REFERENCES

1. Adewoyin, Bunmi & Vassileva, Julita. (2012). Recommendation, Trust and Reputation Management in a Group Online Mentorship System. Kang, M., Yoo, Y. R., & Park, Y. (2012).
2. Analyzing online mentoring process and facilitation strategies. Procedia-Social and Behavioral Sciences, 46, 5158-5162.
3. Nuankaew, P., & Temdee, P. (2015, June). Of online community: Identifying mentor and mentee with compatible different attributes and decision tree. In 2015 12th International Conference on Electrical Engineering/Electronics, Computer,

Telecommunications and Information Technology (ECTI-CON) (pp. 1-6). IEEE.
4. Jin, H. L., & Park, C. J. (2008). Personalized Mentor/Mentee Recommendation Algorithms for Matching in e-Mentoring Systems. The Journal of Korean association of computer education, 11(1), 11-21.
5. Singh, Pradeep & Dutta Pramanik, Pijush & Dey, Avick & Choudhury, Prasenjit. (2021). Recommender Systems: An Overview, Research Trends, and Future Directions. International Journal of Business and Systems Research. 15. 14–52.
6. Van Meteren, R., & Van Someren, M. (2000, May). Using content-based filtering for recommendation. In Proceedings of the machine learning in the new information age: MLnet/ECML2000 workshop (Vol. 30, pp. 47-56).
7. Brun, A., Castagnos, S., & Boyer, A. (2012). Social recommendations: Mentor and leader detection to alleviate the cold-start problem in collaborative filtering. In Social Network Mining, Analysis, and Research Trends: Techniques and Applications (pp. 270-290). IGI Global.
8. Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. In Proceedings of the 10th international conference on World Wide Web (pp. 285-295).
9. Subramaniyaswamy, V., & Logesh, R. (2017). Adaptive KNN based recommender system through mining of user preferences. Wireless Personal Communications, 97(2), 2229-2247.
10. Nagarnaik, P., & Thomas, A. (2015, February). Survey on recommendation system methods. In 2015 2nd International Conference on Electronics and Communication Systems (ICECS) (pp. 1603-1608). IEEE.
11. Breese, J. S., Heckerman, D., & Kadie, C. (2013). Empirical analysis of predictive algorithms for collaborative filtering. arXiv preprint arXiv:1301.7363.
12. Singh, R. H., Maurya, S., Tripathi, T., Narula, T., & Srivastav, G. (2020). Movie recommendation system using cosine similarity and KNN. International Journal of Engineering and Advanced Technology, 9(5), 556-559.
13. Cui, B. B. (2017). Design and implementation of movie recommendation system based on Knn collaborative filtering algorithm. In ITM web of conferences (Vol. 12, p. 04008). EDP Sciences.
14. Taufik, I., Gerhana, Y. A., Ramdani, A. I., & Irfan, M. (2019, December). Implementation K-nearest neighbour for student expertise recommendation system. In Journal of Physics: Conference Series (Vol. 1402, No. 7, p. 077004). IOP Publishing.
15. Bag, S., Ghadge, A., & Tiwari, M. K. (2019). An integrated recommender system for improved accuracy and aggregate diversity. Computers & Industrial Engineering, 130, 187-197.
16. Qian, X., Feng, H., Zhao, G., & Mei, T. (2013). Personalized recommendation combining user interest and social circle. IEEE transactions on knowledge and data engineering, 26(7), 1763-1777.
17. Cho, H., & Ahn, S. A. (2014). An Automated Data-Driven Matching Process for Mentoring Program. In Proceedings of the Korea Contents Association Conference (pp. 227-228). The Korea Contents Association.
18. Haas, C., & Hall, M. (2019). Two-Sided Matching for mentor-mentee allocations—Algorithms and manipulation strategies. Plos one, 14(3), e0213323.
19. Li, C. T. (2019). Mentor-spotting: recommending expert mentors to mentees for live trouble-shooting in Codementor. Knowledge and Information Systems, 61(2), 799-820.
20. Störmann, S., von der Borch, P., Dimitriadis, K., Meinel, F., Reincke, M., & Fischer, M. R.
21. Anggoro, D. A., & Kurnia, N. D. (2020). Comparison of accuracy level of support vector machine (SVM) and K-nearest neighbors (KNN) algorithms in predicting heart disease. *International Journal*, *8*(5), 1689-1694.
22. Al-Shehri, H., Al-Qarni, A., Al-Saati, L., Batoaq, A., Badukhen, H., Alrashed, S., ... & Olatunji, S. O. (2017, April). Student performance prediction using support vector machine and k-nearest neighbor. In *2017 IEEE 30th canadian conference on electrical and computer engineering (CCECE)* (pp. 1-4). IEEE.
23. Fawzy, H., Rady, E. H. A., & Abdel Fattah, A. M. (2020). Comparison between support vector machines and k-nearest neighbor for time series forecasting. *J. Math. Comput. Sci.*, *10*(6), 2342-2359

AUTHORS

Mehak Agrawal received her B.Tech degree in Information Technology from Amity University Uttar Pradesh, Noida, India in 2018. She is currently working as a software engineer in Shell India Market Pvt. Ltd. Company in Bangalore, India. Her areas of interest are artificial intelligence, machine learning and data analytics.

E-mail: mehakagrawal04@gmail.com

Ashir Vijh received his B.Tech degree in Information Technology from Amity University Uttar Pradesh, Noida, India in 2018. He is currently working as a software engineer in Havells India in Noida, India. His areas of interest are machine learning and data analytics.

E-mail: ashirvij2000@gmail.com

Anand Kumar Dubey received his B.Tech degree in Information Technology from Amity University Uttar Pradesh, Noida, India in 2018. He is currently working as a software engineer in PayTM in Noida, India. His areas of interest are machine learning, algorithms, and web development.

E-mail: ananddubey016@gmail.com

Anuranjana received her Doctor of Philosophy degree in Information Technology from Amity University Uttar Pradesh, Noida, India in 2023. She is currently working as an Assistant Professor in Amity University Uttar Pradesh, Noida, India. Her areas of interest are machine learning, artificial intelligence and computer networks.

Corresponding author's E-mail: aranjana@amity.edu