International Journal Software Engineering and Computer Science (IJSECS)
6 (2), 2026, 705-714
Published Online August 2026 in IJSECS (http://www.journal.lembagakita.com/index.php/ijsecs)
P-ISSN: 2776-4869, E-ISSN: 2776-3242. DOI: https://doi.org/10.35870/ijsecs.v6i2.7659.

RESEARCH ARTICLE Open Access

Design of a Student Thesis Topic Recommendation System Using the K-Nearest Neighbor Algorithm

Muhammad Farhan [1*], **Eza Budi Perkasa** [2]

[1*,2] Department of Informatics Engineering, Faculty of Information Technology, Institut Sains dan Bisnis Atma Luhur, Pangkal Pinang City, Bangka Belitung Islands Province, Indonesia.

**Corresponding author*: 2211500067@mahasiswa.atmaluhur.ac.id

Received: June 21, 2026; Accepted: July 3, 2026; Published: August 1, 2026.

Abstract: Determining an appropriate thesis topic can be challenging for students when the selected topic does not align with their skills, interests, and experience, potentially affecting the research process. This study aims to design a data-based thesis topic recommendation system using the K-Nearest Neighbor (KNN) algorithm. Data were collected through a questionnaire measuring three main aspects, namely skills, interests, and experience, across four areas: programming, web development, system security, and computer networks. The questionnaire responses were converted into numerical data using a Likert scale and binary values for classification. The KNN algorithm was implemented using Euclidean Distance and a majority voting mechanism with K values of 3 and 5. The dataset was divided into training and test data using an 80:20 ratio, resulting in an accuracy of 80% on the test data. These results indicate that KNN can be applied to recommend thesis topics based on student characteristics. However, the relatively small dataset means that the findings should be interpreted within the scope of the data used in this study.

Keywords: Recommendation System; Thesis Topic; K-Nearest Neighbor.

1. Introduction

Determining an appropriate thesis topic is one of the challenges faced by students during the preparation of their final research. The selection of a thesis topic is not always aligned with the student's skills, interests, and previous experience. A mismatch between these aspects may make it difficult for students to determine a topic that is suitable for their abilities and areas of interest. Therefore, a recommendation system that considers student characteristics can help provide a more systematic basis for selecting a thesis topic. In academic environments, thesis topic selection is generally influenced by several factors, including students' technical abilities, personal interests, and practical experience. Students may have different levels of competence in programming, web development, system security, and computer networking. They may also have different interests and experiences in these areas. These differences need to be considered because a thesis topic that matches a student's competencies and interests can provide a more appropriate starting point for conducting research. A recommendation system can be used to process these characteristics and generate topic recommendations based on the similarity between student profiles.

Several studies have applied recommendation systems and machine learning methods to provide personalized recommendations in different domains. The K-Nearest Neighbor (KNN) algorithm is one method that can be applied to classification and recommendation problems based on the similarity between data. KNN determines the nearest data points by calculating the distance between the data being evaluated and the training data. One commonly used distance measure is Euclidean Distance. The final classification can then be

determined through majority voting among the nearest neighbors (Bahrani *et al.*, 2023; Khanal *et al.*, 2019; Patro *et al.*, 2020). The relatively simple implementation of KNN also makes it suitable for classification problems involving student profile data. Previous research has demonstrated the use of KNN in recommendation systems. Fahrizal and Hasugian (2025), for example, applied KNN to recommend TV series based on user profiles, while Wijaya *et al.* (2025) implemented TF-IDF and KNN for automatic journal recommendations. These studies show that similarity-based approaches can be applied to generate recommendations according to the characteristics of users. However, the application of KNN for recommending thesis topics based on a combination of students' skills, interests, and experience remains an area that can be further studied. In particular, the use of these three aspects across several areas of informatics competence can provide a profile-based approach to thesis topic recommendation.

Based on this problem, this study aims to design and implement a thesis topic recommendation system using the K-Nearest Neighbor (KNN) algorithm. The system uses questionnaire data representing students' skills, interests, and experience in programming, web development, system security, and computer networking. The data are converted into numerical attributes and classified into four thesis topic categories, namely applications, web development, system security, and computer networks. The KNN algorithm is then applied using Euclidean Distance and majority voting with K values of 3 and 5. The performance of the system is evaluated using an 80:20 division of training and test data and accuracy as the evaluation metric. The results are expected to provide a data-based approach to assist students in selecting thesis topics that correspond to their characteristics.

2. Related Work

This section reviews previous studies related to thesis topic recommendation systems, the application of the K-Nearest Neighbor (KNN) algorithm, and the use of multi-dimensional student profiles in academic recommendation. The review focuses on studies that provide a basis for applying similarity-based classification to recommend topics according to student characteristics.

2.1 K-Nearest Neighbor in Recommendation Systems

The K-Nearest Neighbor (KNN) algorithm is a simple and interpretable machine learning method that can be applied to classification and recommendation problems based on the similarity between data instances. KNN determines the nearest data points to a new instance by calculating the distance between the new instance and the training data. One commonly used distance measure is Euclidean Distance, which calculates the distance based on the differences between corresponding attributes. The class of a new instance can then be determined through majority voting among the selected nearest neighbors (Bahrani *et al.*, 2023; Khanal *et al.*, 2019). Previous studies have applied KNN to recommendation tasks in different domains. Fahrizal and Hasugian (2025) applied KNN to recommend TV series based on user profiles, while Wijaya *et al.* (2025) combined TF-IDF and KNN for automated journal recommendations. These studies indicate that similarity-based approaches can be used to generate recommendations according to the characteristics of users. Patro *et al.* (2020) also examined the use of KNN-based approaches in recommendation tasks and showed the potential of similarity-based methods to identify relevant recommendations from user-related data. The application of KNN in thesis topic recommendation can be based on the similarity between student profiles. In this study, the profile consists of skills, interests, and experience in several areas of informatics, including programming, web development, system security, and computer networks. These attributes are used to represent the characteristics of students and classify them into thesis topic categories. The use of KNN allows the recommendation to be determined from students with similar profile characteristics in the training dataset.

2.2 Multi-Dimensional Student Profiling for Academic Recommendation

Student characteristics are an important consideration in developing academic recommendation systems. A student's selection of a thesis topic may be related not only to academic ability but also to personal interests and previous experience. Therefore, using multiple attributes can provide a broader representation of the student's profile than relying on a single characteristic. Previous educational research has examined the importance of considering multiple dimensions of student characteristics in understanding academic development and student trajectories (Xie *et al.*, 2023; Bowden *et al.*, 2021). Other studies have also discussed the use of multiple factors in educational support and prediction (Stodden *et al.*, 2023; Pavlidou *et al.*, 2021). These studies provide a basis for considering several student characteristics when developing an academic recommendation approach. In this study, student profiles are represented through three main aspects: skills, interests, and experience. Skills and interests are measured using a Likert scale, while experience is represented using binary values. The attributes cover programming, web development, system security, and

computer networks. The resulting numerical representation is then used as input for the KNN classification process.

2.3 KNN, Dataset Size, and Classification Performance

The performance of KNN is affected by the characteristics of the dataset, including the number of observations, the attributes used, and the distribution of the classes. Previous studies have discussed the limitations that can arise when machine learning models are trained using small or unrepresentative datasets (Gong *et al.*, 2023; Grady *et al.*, 2021). These considerations are particularly relevant to academic recommendation systems because the number of available student profiles may be limited. The dataset used in this study consists of 25 student respondents and four thesis topic categories. The data are divided into training and test sets using an 80:20 ratio. The KNN model is evaluated using K values of 3 and 5, with accuracy used to measure the classification performance. The evaluation produces an accuracy of 80% for both K values. Because the test set contains only five observations, the accuracy result is interpreted within the scope of the dataset used in this study rather than as a general measure of KNN performance. Based on the reviewed studies, KNN provides a suitable approach for a thesis topic recommendation system because it can classify new student profiles based on their similarity to existing profiles. However, the limited number of respondents in this study means that the resulting performance should be interpreted cautiously. Further evaluation using a larger dataset and more extensive validation can be considered in future research.

3. Methodology

This study uses a quantitative experimental approach to design and implement a student thesis topic recommendation system using the K-Nearest Neighbor (KNN) algorithm. The approach was used to process numerical data obtained from questionnaires and evaluate the performance of the system in classifying thesis topics based on student characteristics. The research stages consisted of data collection, data preprocessing, dataset construction, KNN implementation, system testing, and performance evaluation.

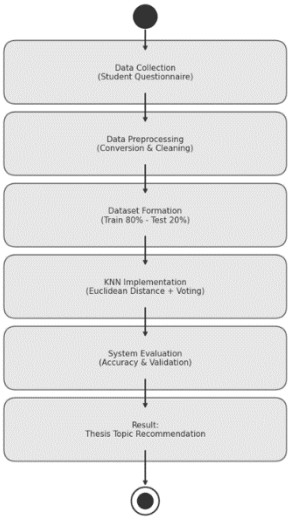

Figure 1. Research Methodology Flow Chart

Figure 1 illustrates the stages used to develop the thesis topic recommendation system using the KNN algorithm. The process begins with data collection through a questionnaire designed to measure three main aspects of students, namely skills, interests, and experience, across four areas of competence. The collected data are then processed and converted into numerical values using a Likert scale and binary coding. The processed data are organized into a structured dataset containing predictor attributes and thesis topic category labels. The dataset is divided into training and test data using an 80:20 ratio. The KNN algorithm is then applied by calculating Euclidean Distance, identifying the nearest neighbors, and determining the predicted class through majority voting. The final stage involves evaluating the classification performance using accuracy.

3.1 Data Collection

The research data were collected through a structured questionnaire distributed to students in the Informatics and Information Systems Engineering study programs. The questionnaire was designed to

measure three main variables, namely skills, interests, and experience, across four areas of competence: programming, web development, system security, and computer networking. Each respondent provided a self-assessment of their competencies based on their academic and practical experience. The dataset consisted of 25 student respondents. The collected responses were used as the basis for constructing the student profile dataset and determining the corresponding thesis topic categories.

3.2 Data Preprocessing

The questionnaire responses were processed into numerical values so that they could be used as input for the KNN algorithm. Skills and interests were measured using a Likert scale ranging from 1 to 5, where 1 represents a very low level and 5 represents a very high level. Experience was represented using binary values, with 1 for a "Yes" response and 0 for a "No" response. Data preprocessing also included removing attributes that were not required for classification, such as respondent names and timestamps. The resulting dataset retained only the attributes relevant to the student profile and thesis topic classification.

3.3 Dataset Construction

Each respondent was represented as one data instance consisting of 12 predictor attributes and one target attribute. The predictor attributes consisted of four skills attributes, four interests attributes, and four experience attributes representing the four areas of competence. The target attribute consisted of four thesis topic categories: 0 for Applications, 1 for Web Development, 2 for System Security, and 3 for Computer Networks. The dataset was divided into training and test data using an 80:20 ratio. Based on the total of 25 respondents, 20 observations were used as training data and 5 observations were used as test data. The training data were used as the reference set for KNN classification, while the test data were used to evaluate the classification results.

3.4 K-Nearest Neighbor (KNN) Algorithm

The K-Nearest Neighbor (KNN) algorithm was implemented as a classification method to determine thesis topic recommendations based on the similarity between student profiles. For each test instance, the distance between the test data and each training instance was calculated. The K nearest instances with the smallest distances were then selected as the nearest neighbors. Euclidean Distance was used to measure the distance between two data vectors using the following formula:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)2} \quad (1)$$

Where x_i and y_i represent the values of the i -th attribute in the two data vectors, and n represents the total number of attributes. This study evaluated two K values, namely K = 3 and K = 5, to examine the classification results produced by different numbers of nearest neighbors. After the nearest neighbors were identified, majority voting was applied. The class with the highest frequency among the selected K nearest neighbors was assigned as the predicted thesis topic.

3.5 System Testing Procedure

System testing was conducted by applying the KNN classification process to the five test instances that had been separated from the dataset. Each test instance was compared with all training instances using Euclidean Distance. The distances were then sorted from the smallest to the largest values to identify the nearest neighbors. For each test instance, the K nearest neighbors were selected using K = 3 and K = 5. The thesis topic category was determined based on the class with the highest frequency among the selected neighbors. The predicted category was then compared with the actual category of the test instance to determine whether the classification was correct. Manual calculations were also used to illustrate the KNN calculation process for the test data. This calculation demonstrates how Euclidean Distance and majority voting were used to determine the recommended thesis topic.

3.6 System Evaluation

System performance was evaluated using accuracy, which measures the proportion of correctly classified test instances relative to the total number of test instances. Accuracy was calculated using the following formula:

$$\text{Accuracy} = \frac{\text{Number of Correct Predictions}}{Total\ Number\ of\ Test\ Data} x\ 100\% \quad (2)$$

The accuracy value was calculated separately for K = 3 and K = 5. The resulting values were used to compare the classification performance of the two K values on the test data. In addition to the numerical accuracy

result, the classification outcomes were examined to determine whether the recommended thesis topic was consistent with the profile attributes of the corresponding student. Because the test set consisted of only five observations, the accuracy results were interpreted within the scope of the dataset used in this study.

4. Result and Discussion

4.1 Results

This study involved 25 students from the Informatics and Information Systems Engineering study program. The respondents were selected using purposive sampling based on the criteria of final-year students who had completed at least 100 credits and were preparing or conducting their thesis. Data were collected through a structured questionnaire measuring three main aspects, namely skills, interests, and experience, across four competency areas: programming, web development, system security, and computer networking. The distribution of respondents by preferred thesis topic consisted of 10 students (40%) for Web Development, 8 students (32%) for Applications, 4 students (16%) for System Security, and 3 students (12%) for Computer Networks. The distribution indicates that the number of observations differs across the four topic categories. This difference should be considered when interpreting the classification results because categories with fewer observations may have less representation in the training data.

Table 1. Distribution of Respondents Based on Thesis Topic

Thesis Topic	Number of Students	Percentage
Web Development	10	40%
Applications	8	32%
System Security	4	16%
Computer Networks	3	12%
Total	25	100%

The respondents were also distributed across semesters 6 to 8, with reported GPA values ranging from 3.0 to 3.8. These characteristics describe the academic background of the respondents. However, semester and GPA were not included among the 12 predictor attributes used in the KNN classification. The questionnaire responses were converted into numerical values before being used in the KNN classification process. Skills and interests were measured using a Likert scale from 1 to 5, where 1 represents a very low level and 5 represents a very high level. Experience was coded using binary values, with 1 representing "Yes" and 0 representing "No." Each respondent was represented by 12 predictor attributes consisting of four skills attributes, four interests attributes, and four experience attributes. The target variable consisted of four thesis topic categories: 0 for Applications, 1 for Web Development, 2 for System Security, and 3 for Computer Networks. The dataset was divided into training and test data using an 80:20 ratio and random sampling. The division resulted in 20 training observations and 5 test observations. The training data were used as the reference data for the KNN classification process, while the test data were used to evaluate the classification results.

Table 2. Dataset Structure Used

Attribute Category	Number of Attributes	Measurement Scale	Example Values
Skills	4	Likert 1–5	3, 4, 5
Interests	4	Likert 1–5	2, 3, 4
Experience	4	Binary 0/1	0, 1
Target Labels	1	Categorical 0–3	0, 1, 2, 3
Total	13	—	—

The KNN system was tested using the five test observations separated from the dataset. Each test observation was compared with the 20 training observations using Euclidean Distance. The observations with the smallest distance values were selected as the nearest neighbors, and the predicted thesis topic was determined through majority voting. Two K values, namely K = 3 and K = 5, were evaluated. For each test observation, the distances to the training observations were calculated and ranked from the smallest to the largest value. The K nearest observations were then used to determine the predicted class through majority voting. For K = 3, the available test results show that four of the five test observations were classified correctly, while one observation was classified incorrectly. Test data 4 was classified as Web Development although its actual label was Computer Networks.

Table 3. System Prediction Results for Test Data with K = 3

Test Data	System Prediction	Actual Label	Status	Reported Closest Distance
1	Web Development	Web Development	Correct	0.85
2	Applications	Applications	Correct	1.12
3	System Security	System Security	Correct	0.95
4	Web Development	Computer Networks	Incorrect	1.45
5	Web Development	Web Development	Correct	0.78

For K = 3, the classification process was based on the three nearest neighbors identified from the training data. The predicted thesis topic was determined through majority voting among these neighbors. Therefore, the results should include the class labels and corresponding distance values of the three nearest neighbors for each test observation. Providing these details allows the classification process and the resulting prediction to be examined more clearly. For K = 5, the system produced four correct predictions and one incorrect prediction, resulting in an accuracy of 80%. To provide a clearer comparison between the two K values, the prediction results for each test observation should also be presented, including the five nearest neighbors, their corresponding class labels and distance values, and the resulting majority vote. This information is necessary to verify how the K = 5 classification results were obtained.

Table 4. Comparison of Results Using Different K Values

K Value	Correct Predictions	Incorrect Predictions	Accuracy
K = 3	4	1	80%
K = 5	4	1	80%

Based on the five test observations, four predictions were classified correctly and one prediction was classified incorrectly. The resulting accuracy was:

$$\text{Accuracy} = \frac{4}{5} x\ 100\% = 80\%$$

The same accuracy was obtained for both K = 3 and K = 5. The incorrect classification occurred in test data 4, where the actual category was Computer Networks but the predicted category was Web Development. This result indicates that the similarity between the profile of the test observation and the training observations did not lead to the correct classification for this particular case. Because only five observations were used as test data, the 80% accuracy should be interpreted within the scope of this dataset. The result provides an initial indication of the classification performance of KNN but is not sufficient to represent the general performance of the system on a larger student population. One factor that may affect the classification results is the unequal distribution of observations across thesis topic categories. Web Development accounted for 40% of respondents, followed by Applications at 32%, System Security at 16%, and Computer Networks at 12%. This distribution means that some categories have fewer observations available as references for the KNN classification process. However, the available results do not by themselves establish that the unequal distribution was the direct cause of the incorrect prediction. A more specific analysis would require examining the class labels and distances of the nearest neighbors for the incorrectly classified test observation. Another limitation is the small dataset size. The study used 25 respondents, with only five observations allocated to the test set. This small test set makes the accuracy value sensitive to individual predictions. Therefore, increasing the number and diversity of respondents would provide a stronger basis for evaluating the generalization performance of the recommendation system. The K value is another factor that may influence KNN classification. A small K value gives greater weight to local patterns, while a larger K value considers a broader set of neighboring observations. In this study, K = 3 and K = 5 produced the same accuracy of 80%. This result indicates that changing the K value within the tested range did not change the overall accuracy for this dataset. The attributes used in the system may also influence the recommendations. The study uses 12 predictor attributes representing skills, interests, and experience across four competency areas. Additional attributes, such as preferred courses or previous project experience, could be considered in future studies if they are shown to be relevant to thesis topic selection.

4.2 Discussion

The results indicate that the K-Nearest Neighbor algorithm can be applied to classify thesis topic categories based on student profile attributes. Using 20 training observations and five test observations, both K = 3 and K = 5 produced an accuracy of 80%. The result is consistent with the general use of KNN in recommendation and classification tasks because the method determines the class of a new observation based on its similarity to existing observations. Previous studies have also applied KNN to recommendation problems

in different domains, including content and academic-related recommendations. The use of Euclidean Distance and nearest-neighbor voting provides a relatively simple approach for processing profile-based data. However, direct comparison of accuracy values between different studies should be made cautiously because the datasets, number of classes, attributes, and evaluation procedures may differ. The use of three student profile dimensions—skills, interests, and experience—provides the basis for representing students through multiple attributes. In this study, these attributes are distributed across programming, web development, system security, and computer networking. This approach allows the recommendation process to consider more than one characteristic when determining a thesis topic category. The results also indicate several limitations. First, the dataset contains only 25 respondents, with five observations used for testing. Second, the distribution of respondents across topic categories is unequal, with Web Development having the largest number of observations and Computer Networks the smallest. These conditions limit the extent to which the 80% accuracy can be generalized to a larger population. The study therefore provides an initial application of KNN for thesis topic recommendation based on student profiles. Further research can use a larger dataset and more extensive evaluation procedures to obtain a more reliable assessment of classification performance. Additional student attributes may also be examined to determine whether they provide useful information for thesis topic recommendation.

5. Conclusion and Recommendations

This study designed and implemented a student thesis topic recommendation system using the K-Nearest Neighbor (KNN) algorithm. The system uses three main aspects of student profiles, namely skills, interests, and experience, across four competency areas: programming, web development, system security, and computer networking. Data were collected through questionnaires from 25 respondents and converted into numerical values using a Likert scale and binary coding. The KNN classification process was evaluated using an 80:20 training-to-test data split with K = 3 and K = 5. Both K values produced an accuracy of 80% on the test data, with four of five test observations classified correctly. The results indicate that KNN can be applied to classify thesis topic categories based on similarities among student profiles. However, the 80% accuracy should be interpreted within the scope of the dataset used in this study because only 25 respondents were involved and five observations were used as test data. Therefore, the findings provide an initial indication of the applicability of KNN for thesis topic recommendation rather than a general measure of system performance.

The dataset is relatively small, and the distribution of respondents across thesis topic categories is uneven, with Web Development having the largest proportion of respondents and Computer Networks having the smallest. These characteristics may affect the classification results and limit the generalization of the findings. In addition, the study only evaluated K = 3 and K = 5, so the effect of other K values was not examined. For future research, the number and diversity of respondents can be increased to provide a broader representation of student profiles. Further studies may also evaluate additional K values, examine class balancing techniques such as oversampling, and investigate additional attributes that may be relevant to thesis topic selection. More extensive validation using a larger dataset can also be conducted to obtain a more reliable evaluation of the system's classification performance. Hybrid recommendation approaches may be considered as an alternative for future development if they are supported by the characteristics and requirements of the expanded dataset.

References

Alhilal, M., & Astuti, H. (2025). Expert system for laptop fault diagnosis using the forward chaining method. *Jurnal Inovasi Ilmu Komputer, 4*(1), 23–34.

Bahrani, P., Minaei-Bidgoli, B., Parvin, H., Mirzarezaee, M., & Keshavarz, A. (2024). A new improved KNN-based recommender system. *The Journal of Supercomputing, 80*(1), 800–834. https://doi.org/10.1007/s11227-023-05447-1

Bowden, J. L. H., Tickle, L., & Naumann, K. (2021). The four pillars of tertiary student engagement and success: A holistic measurement approach. *Studies in Higher Education, 46*(6), 1207–1224. https://doi.org/10.1080/03075079.2019.1672647

Burhanuddin, N. I., & Akhsa, A. T. P. D. (2021). Identifikasi kerusakan laptop dengan metode certainty factor berbasis Android. *Jurnal Teknologi Komputer, 1*, 53–60.

Diranisha, V., Triayudi, A., & Komalasari, R. T. (2024). Implementation of K-nearest neighbour (KNN) algorithm and random forest algorithm in identifying diabetes. *SAGA: Journal of Technology and Information System, 2*(2), 234–244. https://doi.org/10.58905/saga.v2i2.253

Fahrizal, D., & Hasugian, A. H. (2025). Sistem rekomendasi TV series berdasarkan genre menggunakan algoritma KNN. *INSOLOGI: Jurnal Sains dan Teknologi, 4*(4), 895–906. https://doi.org/10.55123/insologi.v4i4.6225

Gong, Y., Liu, G., Xue, Y., Li, R., & Meng, L. (2023). A survey on dataset quality in machine learning. *Information and Software Technology, 162*, 107268. https://doi.org/10.1016/j.infsof.2023.107268

Goyal, S. (2022). Handling class-imbalance with KNN (neighbourhood) under-sampling for software defect prediction. *Artificial Intelligence Review, 55*(3), 2023–2064. https://doi.org/10.1007/s10462-021-10044-w

Grady, C. L., Rieck, J. R., Nichol, D., Rodrigue, K. M., & Kennedy, K. M. (2021). Influence of sample size and analytic approach on stability and interpretation of brain-behavior correlations in task-related fMRI data. *Human Brain Mapping, 42*(1), 204–219. https://doi.org/10.1002/hbm.25217

Jaelani, A., & Akbar, R. (2025). Perancangan sistem pakar berbasis web untuk menentukan kerusakan komputer menggunakan metode certainty factor. *Journal of Computer Science and Information Technology, 1*(1), 26–31.

Kalyzta, J., & Syafrullah, M. (2023). Sistem pakar diagnosa kerusakan komputer dengan algoritma certainty factor pada Lab ICT Budi Luhur pada Universitas Budi Luhur. *[Nama jurnal perlu dilengkapi], 6*, 11–21.

Khan, H. U., Naz, A., Alarfaj, F. K., & Almusallam, N. (2025). A transformer-based architecture for collaborative filtering modeling in personalized recommender systems. *Scientific Reports, 15*(1), 24503. https://doi.org/10.1038/s41598-025-08931-1

Khanal, S. S., Prasad, P. W. C., Alsadoon, A., & Maag, A. (2020). A systematic review: Machine learning based recommendation systems for e-learning. *Education and Information Technologies, 25*(4), 2635–2664. https://doi.org/10.1007/s10639-019-10063-9

Kurnia, M. A. R., & Haidir, A. (2024). Perancangan sistem pakar dalam mendiagnosa kerusakan pada laptop berbasis web menggunakan metode certainty factor. *Journal of System Management and Innovation, 4*(2), 74–83.

Laksana, T. G., Iskandar, A. R., & Ahmad, W. N. W. (2024). Comparison of CPU damage prediction accuracy between certainty factor and forward chaining techniques. *Jurnal Pekommas, 9*, 109–119. https://doi.org/10.56873/jpkm.v9i1.5531

Li, J., Chi, X., Ji, X., & Yao, Y. (2025). Research on human resource matching algorithm based on collaborative filtering and learning algorithm. In *2025 IEEE International Conference on Networks, Multimedia and Information Technology (NMITCON 2025)*. https://doi.org/10.1109/NMITCON65824.2025.11189014

Low, D. M., Bentley, K. H., & Ghosh, S. S. (2020). Automated assessment of psychiatric disorders using speech: A systematic review. *Laryngoscope Investigative Otolaryngology, 5*(1), 96–116. https://doi.org/10.1002/lio2.354

Marpaung, A. J., & Handoko, K. (2023). Sistem Pakar Untuk Diagnosa Kerusakan Komputer Menggunakan Metode Forward Chaining Dan Certainty Factor Berbasis Web. *Computer and Science Industrial Engineering (COMASIE)*, *9*(6).

Marsandi, A. F., Pratama, A. R., Kusumaningrum, D. S., & Rohana, T. (2024). Sistem pakar deteksi kerusakan laptop menggunakan algoritma forward chaining dan backward chaining. *KLIK: Kajian Ilmiah Informatika dan Komputer, 5*(1), 49–56. https://doi.org/10.30865/klik.v5i1.2041

Mohammed, R., Rawashdeh, J., & Abdullah, M. (2020). Machine learning with oversampling and undersampling techniques: Overview study and experimental results. In *2020 11th International Conference on Information and Communication Systems (ICICS)* (pp. 243–248). https://doi.org/10.1109/ICICS49469.2020.239556

Mujahid, M., Kına, E. R. O. L., Rustam, F., Villar, M. G., Alvarado, E. S., De La Torre Diez, I., & Ashraf, I. (2024). Data oversampling and imbalanced datasets: An investigation of performance for machine learning and feature engineering. *Journal of Big Data, 11*(1), 87. https://doi.org/10.1186/s40537-024-00943-4

Nagarkar, G., & Savyanavar, A. (2025). Fusion of NLP and nutritional intelligence for food recommendation: A three-way hybrid model. In *2025 International Conference on Next Generation Computing Systems: Intelligent System for Sustainable Development (ICNGCS 2025)*. https://doi.org/10.1109/ICNGCS64900.2025.11183285

Park, C., Awadalla, A., Kohno, T., Patel, S., & Allen, P. G. (2021). Reliable and trustworthy machine learning for health using dataset shift detection. In *Advances in Neural Information Processing Systems, 34*, 3043–3056.

Patro, S. G. K., Mishra, B. K., Panda, S. K., Kumar, R., Long, H. V., Taniar, D., & Priyadarshini, I. (2020). A hybrid action-related K-nearest neighbour (HAR-KNN) approach for recommendation systems. *IEEE Access, 8*, 90978–90991. https://doi.org/10.1109/ACCESS.2020.2994056

Pavlidou, I., Dragicevic, N., & Tsui, E. (2021). A multi-dimensional hybrid learning environment for business education: A knowledge dynamics perspective. *Sustainability, 13*(7), 3889. https://doi.org/10.3390/su13073889

Permadi, D., Handayani, D., Kustanto, P., Yasir, M., Noeman, A., & Hidayat, A. (2025). Implementasi forward chaining pada sistem pakar diagnosa kerusakan laptop berbasis web: Studi kasus layanan servis laptop. *Jurnal Information and Information Security, 6*(2), 115–128.

Pratama, H. S., Efendy, M. P., Roby, M., & Tusakdiyah, S. H. (2022). Sistem Pakar Deteksi Kerusakan Laptop Atau Komputer Menggunakan Metode Forward Chaining. *Jurnal Teknik Informatika*, *2*(1).

Rendón, E., Alejo, R., Castorena, C., Isidro-Ortega, F. J., & Granda-Gutiérrez, E. E. (2020). Data sampling methods to deal with the big data multi-class imbalance problem. *Applied Sciences, 10*(4), 1276. https://doi.org/10.3390/app10041276

Saputra, O., Fitri, I., & Handayani, E. T. E. (2022). Sistem pakar diagnosa kerusakan hardware komputer menggunakan metode forward chaining dan certainty factor berbasis website. *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi), 6*(2), 0–8.

Simatupang, R. B. J., & Pusparini, N. N. (2025). Sistem pakar diagnosa kerusakan notebook berbasis web menggunakan metode certainty factor dengan pengujian blackbox. *Jurnal Riset Sistem Informasi dan Teknik Informatika, 3*, 247–266.

Stodden, D. F., Pesce, C., Zarrett, N., Tomporowski, P., Ben-Soussan, T. D., Brian, A., Abrams, T. C., & Weist, M. D. (2023). Holistic functioning from a developmental perspective: A new synthesis with a focus on a multi-tiered system support structure. *Clinical Child and Family Psychology Review, 26*(2), 343–361. https://doi.org/10.1007/s10567-023-00428-5

Trstenjak, B., Mikac, S., & Donko, D. (2014). KNN with TF-IDF based framework for text categorization. *Procedia Engineering, 69*, 1356–1364. https://doi.org/10.1016/j.proeng.2014.03.129

Widianto, S. C., Widada, B., Sandradewi, K., & Remawati, D. (2025). Implementasi metode certainty factor dalam sistem pakar untuk diagnosa kerusakan laptop. *Jurnal TIKomSiN, 13*(1), 40–49.

Wijaya, J., Putra, F. S., Irsyad, H., & Rahman, A. (2025). Implementasi TF-IDF dan KNN pada rekomendasi jurnal otomatis. *Journal of Informatics and Computer Engineering Research, 2*(1), 18–24. https://doi.org/10.31963/jicer.v2i1.5565

Xie, K., Vongkulluksn, V. W., Heddy, B. C., & Jiang, Z. (2024). Experience sampling methodology and technology: An approach for examining situational, longitudinal, and multi-dimensional characteristics of engagement. *Educational Technology Research and Development, 72*(5), 2585–2615. https://doi.org/10.1007/s11423-023-10259-4