Machine Learning-Driven Medical Recommendation System for Early Disease Prediction and Personalized Treatment

Gaurav Singh Negi[1], Surya Kant Pal[1], Utpal Dhar Das[1], Saloni Srivastava[1], Hari Shankar Shyam[2]

[1]Department of Mathematics & Data Science, SSES, Sharda University, Greater Noida-201310, UP, India

[2]Sharda School of Business Studies, Sharda University, Greater Noida, Uttar Pradesh, India

(e-mail Id: negigaurav746@gmail.com; suryakantpal6676@gmail.com; utpal.statmath@gmail.com; salonisrivastava007@gmail.com; harishankar.shyam@sharda.ac.in)

*Corresponding Author: suryakantpal6676@gmail.com

Abstract

This project presents an intelligent Naive Bayes and K-Nearest Neighbors (KNN) based medical recommendation system to diagnose a disease based on symptoms provided by patients and prescribe personalized medicine. Using effective strategies such as feature encoding and balancing data; the system operates with high efficiency across different clinical data sets. Experimental results show the Naive Bayes model attains a striking accuracy rate of 96% compared to KNN and popular algorithms such as Random Forest because it operates smoothly with categorical features. The system not only efficiently diagnoses disease patterns with satisfactory accuracy but also effectively provides recommendations such as precautions, medicines, and lifestyle change and assists doctors with a robust decision- support system. This project explains how machine learning can enhance accuracy in diagnosis, reduce human mistakes, and benefit patients and how it provides future directions in developing AI-based health care solutions.

Keywords: K-Nearest Neighbors, RandomForest, Robust, Healthcare, Accuracy

1 Introduction

Healthcare has reached a stage where new technology can revolutionize the way

R. Nagariya et al. (eds.), *Proceedings of the International Conference on Sustainable Business Practices and Innovative Models (ICSBPIM-2025)*, Advances in Economics, Business and Management Research 349,
https://doi.org/10.2991/978-94-6463-872-1_63

we identify and treat diseases in patients. These problems indicate that there is a significant requirement for data-driven tools that enable doctors to make faster and better decisions. This study proposes a machine learning system that can identify diseases early and recommend customized treatment plans based on input symptoms. The system employs advanced techniques, like K-Nearest Neighbors (KNN) and Naive Bayes, to examine patient data and provide good medical recommendations. KNN employs similar cases to identify matching patterns of symptoms, while Naive Bayes employs prob- ability to estimate the likelihood of a disease, providing both techniques strengths that enhance overall accuracy. Using preprocessing techniques such as feature encoding and data balancing, the system overcomes common issues such as noisy data and imbalanced class distributions, ensuring it performs optimally. The primary objectives of this study are three: to develop an effective machine learning model for disease classification, to evaluate how effective KNN and Naive Bayes are in healthcare, and to demonstrate how AI can improve patient care.

The K-Nearest Neighbors (KNN) algorithm was first conceptualized by Evelyn Fix and Joseph Hodges in 1951 as a non-parameter-dependent classifier. KNN is a learning method that makes predictions based on the most common class among the nearest examples. KNN is highly beneficial in pattern recognition, recommendation, and disease diagnosis because it is simple and flexible. KNN works by calculating the distance (most often Euclidean distance) from the test point to every training case. It then chooses the K nearest neighbors and marks the new case with the most common class among them. This works well on small to medium-sized data but is computationally expensive when working with large data.

Naïve Bayes classifier is derived from Bayes' Theorem, which was developed by 18th-century statistician and theologian Thomas Bayes. Naïve Bayes algorithm was employed for object classification during the 1950s and 1960s and was widely applied for spam filtering, text classification, and medical diagnosis. This algorithm calculates the probability of each category from a given set of input features (e.g., symptoms in medical cases) and selects the most likely category. It is fast, inexpensive, and can work with categorical data, making it suitable for medical recommendation systems.

This project is significant as it introduces new technology to the field of medicine. By making disease detection easier and providing recommendations for every patient, the system minimizes diagnostic errors, accelerates decision-making, and provides expert counsel to more individuals even in times of scarcity. This paper describes how the approaches work, from data preparation, model choice, up to success measurement, and experimental results with tests demonstrating the efficacy of the system. Finally, this study tries to enhance our knowledge of healthcare with AI, to enable intelligent solutions that put patients' health first and utilize medical work more effectively.

2 Literature review

Yang et al [1] provided a comprehensive overview of the latest advancements in recommender systems [1]. In order to make user-experience-based content recommendations for films, products, or other media, the study examined the value of recommendation systems and explained how they might be applied. The researchers discussed a range of recommendation methods, including collaborative filtering, content-based filtering, and hybrid models. They noted that a common technique in their work was collaborative filtering, which suggested items based on user behaviour. This study offered valuable insights into the creation and application of recommender systems across multiple domains.

Pal et al [2] explored how machine learning could be utilized to detect unsolicited mail communications, such as spam emails[2] One of the key findings derived from the research paper was the application of classification algorithms, which aided in distinguishing between symptoms and diseases. These techniques encompassed Naive Bayes Classification, K-Nearest Neighbors, and Random Forest. Overall, the study offered valuable insights into how AI and machine learning were implemented using the Python programming language.

Kiru et al. [11] used deep learning and NLP which became increasingly important in healthcare, especially for tasks like disease prediction and answering patient queries. They also used the tools like TF-IDF, Hugging Face Transformers, PyTorch, and Scikit-learn, and reported strong performance-based metrics like accuracy, precision, recall, F1-score. By doing so, it aimed to support both medical professionals and patients through a reliable, adaptive, and east-to-use system.

Ko et al [3] examined a thorough analysis of recommendation systems, going into detail about their models, methods, and practical uses [3]. On websites like Netflix, Amazon, and Spotify, recommendation systems are frequently used to make product, movie, or music recommendations based on user preferences. The researchers also emphasised the difficulties in implementing recommendation systems by highlighting problems with scalability, bias in recommendations, and data privacy.

Tran et al [4] investigated how recommender systems were employed in the healthcare sector to assist doctors and patients in making informed medical decisions[4]. These systems, which were previously commonly applied in e-commerce and entertainment, emerged as vital tools in healthcare by suggesting treatments, medications, and dietary plans. The research also emphasized the applications of recommender systems in healthcare and identified challenges, such as the necessity of understanding biology-related terminology, which impacted their effective implementation.

Valdiviezo-Diaz et al. [5] proposed a collaborative filtering recommendation system based on a Naïve Bayes classification algorithm [5]. Their work aimed to increase the prediction accuracy of user preferences by fusing probabilistic modelling with collaborative filtering techniques. They demonstrated that the Naïve Bayes model enhanced scalability and more effectively handled sparsity issues when compared to conventional collaborative filtering techniques.

Lambay and Mohideen [6] investigated the application of big data analytics to recommendation systems for healthcare [6]. They discussed how to make better, more individualized health recommendations by utilizing the enormous amount of healthcare data. Their study emphasized the challenges of processing and analyzing healthcare data and the importance of machine learning, security, and efficient data management strategies to enhance healthcare delivery.

Altayeva et al. [7] developed a medical decision-making diagnosis system that integrated k-means clustering with the Naïve Bayes algorithm [7]. They used k-means to cluster patient data into meaningful groups and subsequently applied Naïve Bayes to classify diseases within these clusters. Their approach improved diagnostic accuracy by combining the strengths of both unsupervised and supervised learning methods, offering a reliable tool for assisting clinical decisions.

Repaka et al. [8] designed and implemented a heart disease prediction system using the Naïve Bayes algorithm [8]. Their model focused on analyzing medical datasets to predict the likelihood of heart disease in patients. Through experimental validation, they showed that Naïve Bayes provided a simple yet effective solution for early prediction, highlighting its capability to manage uncertainty and incomplete data in medical records.

Langarizadeh et al. [9] conducted a systematic review of studies applying Naïve Bayesian networks to disease prediction [9]. They found that Naïve Bayes models were widely used due to their simplicity and relatively high predictive performance across various types of diseases. However, they also pointed out inherent limitations, such as the independence assumption among features, which could affect model reliability in complex healthcare scenarios.

Tran et al. [10] provided a comprehensive review of recommender systems in the healthcare domain. They discussed existing methodologies, current challenges, and open research issues [10]. Their findings underscored the importance of personalization, data privacy, and interpretability in healthcare recommenders. Additionally, they emphasized the necessity for future research to tackle biases, enhance transparency, and integrate diverse healthcare data sources.

3 Methodology

The Medical Recommendation System is designed to assist users in identifying potential health issues by analyzing the symptoms they provide. It uses machine learning techniques – specially the KNN algorithm and the Naïve Bayes classifier – to make predictions about possible disease. The overall process involves collecting relevant data, preparing it for analysis, choosing appropriate models, training them, evaluating their accuracy, and finally, deploying the system for real-world use.

Implementation using K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a simple yet effective instance-based learning algorithm used for both classification and regression tasks, It is non-parametric, meaning it doesn't make any assumptions about the underlying data distribution. It allocates new instances to its neighbor's majority class in feature space. Determine the distance between each new data point and every point in the training data set. Using the calculated distance, find the K closest points. Select the most common class label among the K neighbors.

Euclidean Distance: One commonly used measure in KNN is Euclidean Distance, which computes the minimum straight-line distance between two points.

$$d = \sqrt{((x_2 - x_1)^2 + (y_2 - y_1)^2)}$$

- (x_1,y_1) and (x_2,y_2) are two points in a 2-dimensional space.
- $(x_2-x_1)^2$ is the squared difference in the **x-coordinates**.
- $(y_2-y_1)^2$ is the squared difference in the **y-coordinates**.

Once K neighbors have been identified, majority voting is employed by KNN for classifying a new point.

Figure 1 depicting symptom data imported from Excel into Python. The data set consists of a number of rows and columns. The model has been trained on this data set for analysis and forecasting diseases based on user queries. Since there were not any missing values, data cleaning was not needed.

	itching	skin_rash	nodal_skin_eruptions	continuous_sneezing	shivering	chills	joint_pain	stomach_pain	acidity	ulcers
0	1	1	1	0	0	0	0	0	0	
1	0	1	1	0	0	0	0	0	0	
2	1	0	1	0	0	0	0	0	0	
3	1	1	0	0	0	0	0	0	0	
4	1	1	1	0	0	0	0	0	0	

Fig. 1 Data set

Table 1 displays the number of actual and predicted values for a confusion matrix, where TN stands for True Negatives. Variables that have been accurately categorised as negatives are involved in TN. True Positives, or TPs for short, are accurately classified positives. False Positives, or FPs for short, are positive results that are negative. False Negatives, or FNs, are positive values that are mistakenly classified as negatives.

		Predicted	
Actual		Negative	Positive
	Negative	TN	FP
	Positive	FN	TP

Table 1: Number of actual and predicted values for confusion matrix

The accuracy of models is evaluated using a confusion matrix and is determined using the following formula:

$$\text{Accuracy} = TN + TPTN + FP + FN + TP$$

Accuracy, though, is not a trustworthy measurement when dealing with unbalanced da- tasets. Therefore, one should test the data properly and have accurate predicted and ac- tual values for obtaining a valid accuracy rate. Applying KNN algorithm in machine learning to the dataset to find the accuracy of the model.

```
knn_model = KNeighborsClassifier(n_neighbors=150, weights='uniform
knn_model.fit(X_train, y_train)
# Predictions
y_pred = knn_model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)*100
cm = confusion_matrix(y_test, y_pred)
print("Classification Report:")
print(classification_report(y_test, y_pred))
print("Confusion Matrix:")
print(cm)
print(f"KNN Accuracy: {accuracy:.4f}")
```

Fig. 2 KNN Classification

```
Classification Report:
              precision    recall  f1-score   support

           0       0.96      1.00      0.98        24
           1       0.94      0.94      0.94        18
           2       1.00      0.94      0.97        18
           3       1.00      1.00      1.00        23
           4       0.95      1.00      0.98        21
           5       1.00      0.96      0.98        24
           6       0.96      0.92      0.94        24
           7       0.95      1.00      0.98        20
           8       0.97      1.00      0.98        30
           9       1.00      0.83      0.90        23
          10       1.00      1.00      1.00        17
          11       0.92      1.00      0.96        23
          12       0.92      1.00      0.96        23
          13       1.00      0.95      0.98        22
          14       0.97      0.90      0.93        31
          15       0.93      0.78      0.85        18
          16       0.95      1.00      0.98        21
          17       1.00      0.92      0.96        26
          18       0.92      1.00      0.96        24
          19       0.95      0.95      0.95        22
          20       1.00      1.00      1.00        20
          21       0.85      0.97      0.90        29
          22       1.00      0.96      0.98        23
          23       0.95      1.00      0.98        20
          24       1.00      1.00      1.00        21
          25       0.96      0.93      0.94        27
          26       1.00      1.00      1.00        23
          27       0.95      1.00      0.98        20
          28       0.93      1.00      0.97        28
          31       0.97      1.00      0.98        28
          32       0.96      0.96      0.96        28
          33       1.00      1.00      1.00        24
          34       0.96      1.00      0.98        23
          35       0.97      0.94      0.95        31
          36       1.00      1.00      1.00        29
          37       1.00      0.94      0.97        32
          38       1.00      0.97      0.98        29
          39       1.00      0.96      0.98        25
          40       1.00      0.96      0.98        26

    accuracy                           0.97       984
   macro avg       0.97      0.97      0.97       984
weighted avg       0.97      0.97      0.97       984

Confusion Matrix:
[[24  0  0 ...  0  0  0]
 [ 0 17  0 ...  0  0  0]
 [ 0  0 17 ...  0  0  0]
 ...
 [ 0  0  0 ... 28  0  0]
 [ 0  0  0 ...  0 24  0]
 [ 0  0  0 ...  0  0 25]]
KNN Accuracy: 96.6463
```

Fig. 3 Confusion Matrix, Precision, Recall, f-1 Score

Implementation using Naïve's Bayes Classification

The rationale behind the use of multinomial method in this research is that it makes it easy to analyze text data (categorical data). It makes prediction on the chances that a particular event will occur based on the past experience under conditions surrounding such an event.

In this model, Naïve Bayes will predict, for a given set of symptoms, the most probable disease based on selecting the disease with maximum probability. After exploring a number of possibilities, we applied Naïve Bayes on a pre-split training and test set. It was utilized for detecting disease based on symptoms due to the importance of feature selection for classification.

Evaluation

K-Nearest Neighbor, Random Forest, and Gradient Boosting were tried out using the same data also. Naïve Bayes was employed also. Once a model was selected, its per- for- mance was compared against other models and experimented with. Naïve Bayes, with a 96.94% rate of accuracy, was employed to carry out this study. Naïve Bayes was pre- ferred due to being simpler to implement, being time-efficient, and less costly to com- pute relative to other machine algorithms that require higher processing time and space.

```
# using Multi nomial bayes classification
classifier=SVC(kernel='rbf',random_state=0)
classifier.fit(X_train,y_train)
print(classifier.score(X_test,y_test)*100)
```

```
96.95121951219512
```

```
pred=classifier.predict(X_test)
print(classification_report(y_test,pred))
print()
print("Confusion Matrix: \n",confusion_matrix(y_test,pred))
print("Accuracy: \n",accuracy_score(y_test,pred)*100)
```

Fig.4 Naïve's Bayes Classification

Fig. 5 Explain Precision, Recall, and other metrics of evaluation were recorded for this data set. This class report indicates the model's performance. Precision column indicates number of cases correctly classified, and precision value of 97 for all classes indicates that there are no false positives. Recall indicates number of cases correctly classified, and value of 97 indicates that there were some false negative. F1-score is harmonic mean of precision and recall, and with value of 97. Support indicates number od actual instances per class in test set; there were 96 test samples of class 0, for instance. The confusion matrix indicates the performance of classification model. Actual classes are in each row, and each column indicates predicted classes. The entries along the main diagonal (24,16,17,28,24,24) indicate perfect classification for each class. There are no misclassification because all off-diagonal entries are 0.

	precision	recall	f1-score	support
0	1.00	1.00	1.00	24
1	0.94	0.89	0.91	18
2	1.00	0.94	0.97	18
3	1.00	1.00	1.00	23
4	0.88	1.00	0.93	21
5	1.00	0.96	0.98	24
6	0.96	0.92	0.94	24
7	0.95	1.00	0.98	20
8	0.97	1.00	0.98	30
9	1.00	0.91	0.95	23
10	1.00	1.00	1.00	17
11	0.92	1.00	0.96	23
12	0.92	1.00	0.96	23
13	1.00	0.95	0.98	22
14	1.00	0.90	0.95	31
15	0.89	0.94	0.92	18
16	0.95	1.00	0.98	21
17	1.00	0.96	0.98	26
18	1.00	1.00	1.00	24
19	0.95	0.95	0.95	22
20	1.00	1.00	1.00	20
21	0.87	0.93	0.90	29
22	1.00	0.96	0.98	23
23	0.95	1.00	0.98	20
24	1.00	1.00	1.00	21
25	0.96	0.93	0.94	27
26	1.00	1.00	1.00	23
27	1.00	0.95	0.97	20
28	0.90	1.00	0.95	28
29	0.94	0.94	0.94	16
30	0.97	1.00	0.98	30

```
          31       0.97      1.00      0.98        28
          32       0.96      0.96      0.96        28
          33       1.00      1.00      1.00        24
          34       0.96      1.00      0.98        23
          35       0.97      0.94      0.95        31
          36       1.00      1.00      1.00        29
          37       1.00      0.97      0.98        32
          38       1.00      0.97      0.98        29
          39       1.00      0.96      0.98        25
          40       1.00      0.92      0.96        26

    accuracy                           0.97       984
   macro avg       0.97      0.97      0.97       984
weighted avg       0.97      0.97      0.97       984


Confusion Matrix:
 [[24  0  0 ...  0  0  0]
 [ 0 16  0 ...  0  0  0]
 [ 0  0 17 ...  0  0  0]
 ...
 [ 0  0  0 ... 28  0  0]
 [ 0  0  0 ...  0 24  0]
 [ 0  0  0 ...  0  0 24]]
Accuracy:
 96.95121951219512
```

Fig. 5 **Confusion Matrix, Precision, Recall, f-1 Score**

Visualizing Symptom Frequency

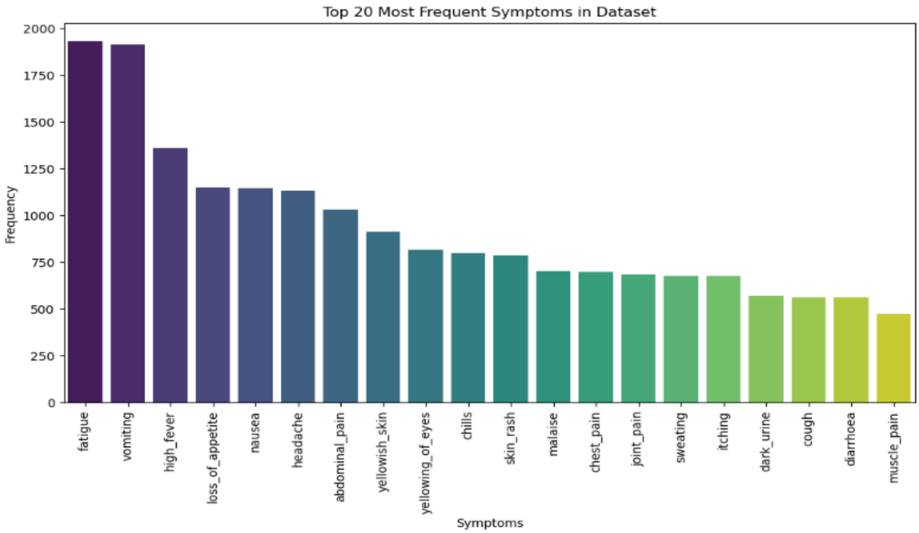

Fig. 6 **most common symptoms in the dataset**

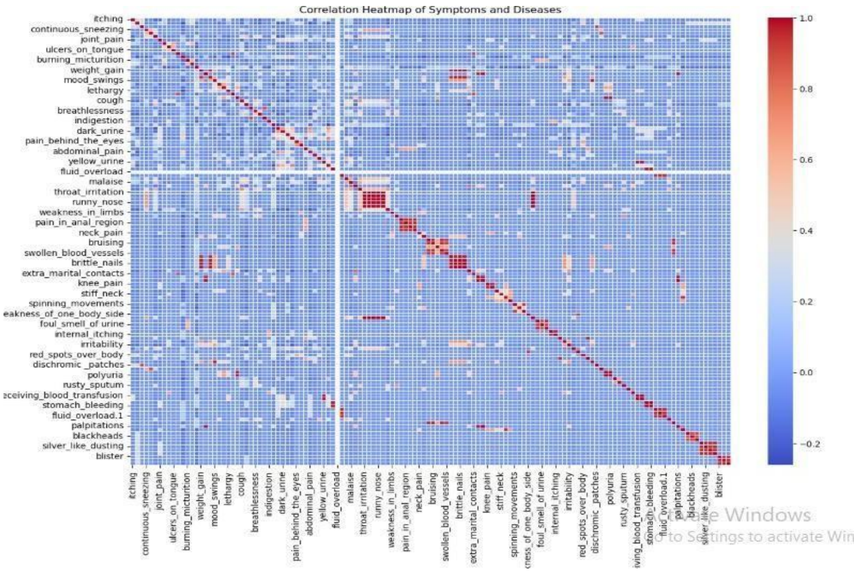

Fig. 7 Correlation Heatmap

4 Results and Outcomes

The goal of the medical recommendation system is to predict illnesses based on the user's entered symptoms and offer relevant medical advice. It processes input symptoms and determines the most likely disease using machine learning algorithms such as Random Forest, Naïve Bayes, and K-Nearest Neighbours (KNN). There are several essential components that make up the system output.

```
Enter your symptoms.......itching
=================predicted disease============
Fungal infection
=================description=================
Fungal infection is a common skin condition caused by fungi.
=================precautions=================
1 :  bath twice
2 :  use detol or neem in bathing water
3 :  keep infected area dry
4 :  use clean cloths
=================medications=================
5 :  ['Antifungal Cream', 'Fluconazole', 'Terbinafine', 'Clotrimazole', 'Ketoconazole']
=================workout=================
6 :  Avoid sugary foods
7 :  Consume probiotics
8 :  Increase intake of garlic
9 :  Include yogurt in diet
10 :  Limit processed foods
11 :  Stay hydrated
12 :  Consume green tea
13 :  Eat foods rich in zinc
14 :  Include turmeric in diet
15 :  Eat fruits and vegetables
=================diets=================
16 :  ['Antifungal Diet', 'Probiotics', 'Garlic', 'Coconut oil', 'Turmeric']
```

5 Conclusion

One of the newest and most widely used techniques for gleaning additional information about a patient from medical data is the health recommendation system. Based on the user-inputted symptoms, this system makes medical recommendations and predicts diseases. In order to identify the most likely disease, this system looks at the symptoms and uses learnt models, such as K- Nearest Neighbors (KNN) or Naïve Bayes.

We intend to refine our algorithm further to deliver enhanced accuracy while maintaining privacy in future releases. In future, they will have to overcome obstacles pertaining to ethics, data privacy, and model interpretability while seizing new opportunities brought about by improvements in data collecting, AI explainability, and telehealth integration. If the system detects a disease, it also provides valuable suggestions, such as precautions to be taken, potential medicines, diet schedules, and exercise routines to get healthier.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

[1]. Li, Yang, Kangbo Liu, Ranjan Satapathy, Suhang Wang, & Erik Cambria. Recent developments in recommender systems: A survey. *IEEE Compu- tational Intelligence Magazine 19(2),* 78- 95. https://ieeexplore.ieee.org/abstract/document/10494051/

[2]. .Pal, Surya Kant, Oma Junior Raffik, Rita Roy, and Prem Shankar Jha. Machine Learning Methodology for the Recognition of Unsolicited Mail Communications. In *International Conference on Advances in IoT and Security with AI*, pp. 51-60. Singapore: Springer Nature Singapore, 2023. https://link.springer.com/chapter/10.1007/978-981-99-5085-0_6

[3]. Ko, Hyeyoung, Suyeon Lee, Yoonseo Park, and Anna Choi. A survey of recommendation systems: recommendation models, techniques, and application fields. *Electronics 11(1),* 141. https://www.mdpi.com/2079-9292/11/1/141

[4] Kabiru Uba Kiru, S. K. Pal, S. S. Ahmad, and I. M. Kabir, Deep Learning-Enhanced Medical System for Disease Prediction, Medication Recommendation, and Intelligent Question-Answering Based on Patient Symptomatology. In *2021 IEEE 8th Uttar Pradesh Section International Conference on Electrical, Electronics and Computer Engineering (UPCON), pp (1–8), Nov.2024.*
https://doi.org/10.1109/upcon62832.2024.10982945.

[5]. Tran, Thi Ngoc Trang, Alexander Felfernig, Christoph Trattner, and An- dreas Holzinger. Recommender systems in the healthcare domain: state- of-the-art and research issues. *Journal of Intelligent Information Systems 57(1), (2021):* 171-201. https://link.springer.com/arti- cle/10.1007/s10844-020-00633-6

[6]. Valdiviezo-Diaz, Priscila, Fernando Ortega, Eduardo Cobos, and Raúl Lara-Cabrera. A collaborative filtering approach based on Naïve Bayes classifier. *IEEE Access 7 (2019):* 108581-108592. https://ieeexplore.ieee.org/abstract/document/8787761/

[7]. Lambay, M. A., & Mohideen, S. P. (2020, July). Big data analytics for healthcare recommendation systems. In 2020 *International Conference on System, Computation, Automation and Networking (ICSCAN)* (pp. 1-6). IEEE https://www.mdpi.com/2071-1050/14/5/2500

[8]. Altayeva, A., Zharas, S., & Im Cho, Y. (2016, October). Medical deci- sion-making diagnosis system integrating k-means and Naïve Bayes algo- rithms. In 2016 *16th international conference on control, automation and systems (ICCAS)* (pp. 1087-1092). IEEE. https://ieeexplore.ieee.org/ab- stract/document/7832446/

[9]. Repaka, A. N., Ravikanti, S. D., & Franklin, R. G. (2019, April). Design and implementing heart disease prediction using naives Bayesian. In 2019 *3rd International conference on trends in electronics and informatics (ICOEI)* (pp. 292-297). IEEE. https://ieeexplore.ieee.org/abstract/docu- ment/8862604/

[10]. Langarizadeh, M., & Moghbeli, F. (2016). Applying naive bayesian net- works to disease prediction: a systematic review. *Acta Informatica Medica, 24(5),* 364. https://pmc.ncbi.nlm.nih.gov/articles/PMC5203736/

[11]. Tran, T. N. T., Felfernig, A., Trattner, C., & Holzinger, A. (2021). Rec- ommender systems in the healthcare domain: state-of-the-art and research issues. *Journal of Intelligent Information Systems, 57(1),* 171-201https://link.springer.com/article/10.1007/s10844-020-00633-6