



Medicine Recommendation System Using Effective Machine Learning Techniques & Predict

¹, Akash Bhardwaj, ¹.Pawan Kumar, ¹ Aditya Tiwari, ¹Abhineet Suman Department of Computer Science
and Engineering (AI&ML)

Nitra Technical Campus, Ghaziabad, 201002, India

Abstract: This paper presents a machine learning-based healthcare recommendation system designed to provide personalized medical advice by accurately predicting diseases from patient symptoms. The system utilizes a comprehensive symptom–disease dataset, leveraging support vector classifier (SVC) and random forest (RF) models, achieving outstanding accuracies of 93.75% in disease prediction. These results surpass those of similar studies, such as multinomial naive techniques, which achieved 91% accuracy but relied on smaller datasets with limited diversity. The proposed system not only excels in diagnosis but also integrates tailored recommendations, including medication, dietary plans, and exercise regimens, to address the specific needs of patients. These personalized recommendations enhance practical utility, offering a patient-centered approach that promotes proactive health management. By focusing on diseases with high and moderate predictive performance, the system addresses both common and complex conditions effectively. The study demonstrates the transformative potential of machine learning in developing scalable and efficient healthcare systems, bridging the gap between accurate prediction and actionable treatment strategies. Future research will aim to incorporate larger and more diverse datasets, address underrepresented diseases, and refine feature engineering to enhance model generalizability and the system's overall effectiveness.

Index Terms - Machine Learning, Medicine Recommendation, Healthcare AI, Personalized Medicine, Decision Support System

I. INTRODUCTION

The healthcare industry is evolving rapidly, and personalized treatments are becoming essential. traditional medicine prescription methods rely heavily on the doctor's expertise and experience, which can vary across individuals and situations. Machine Learning (ML) offers the potential to standardize and optimize the recommendation of medicines based on data-driven insights. This paper discusses the development and evaluation of a Medicine Recommendation System using ML models trained on patient records to suggest accurate medications. The Medicine Recommendation System using Machine Learning is a revolutionary approach aimed at improving healthcare delivery by offering personalized and accurate medication recommendations.

With the rapid advancement of technology, machine learning has shown significant promise in transforming various industries, and healthcare is no exception. In the realm of medicine, selecting the right medicine for a patient can be a complex process due to the wide array of available medications and varying patient conditions. Traditional methods often rely on a physician's experience and medical guidelines, which, while effective, may not always account for the vast individual variability seen in patients. This system leverages machine learning techniques to automate and enhance the process of recommending medications by

considering a variety of factors such as symptoms, medical history, age, gender, allergies, and other personalized information.

Machine Learning Methods and their Application Machine learning is the part of artificial intelligence that emphasis on the learning of the system through training on data sets and developing experience without being commutated by a mathematical expression. The machine learning algorithm can be applied if the pattern exists among the training data sets, we cannot pin down the target function mathematically, there is sufficient data for the system. The different categories of Machine learning are as listed: Supervised Machine Learning: In this type of learning algorithm, the training data sets provided to form the Learning Model for predicting the future events are labelled (i.e., provided with the output). The framework can alter learning by focusing on new contributions. It is the better model used to find errors and better yield.

Unsupervised learning this algorithms are used to prepare the learning model with ungrouped or unlabeled data points. The main essence of this learning is to portray a concealed structure from unlabeled data. It does not make a correct result approximation; however, it helps in extracting major information and deductions about the learning model from the unlabeled data sets. 1.5 Semi-supervised Machine Learning: This learning algorithm falls in the middle of supervised and unsupervised learning since this makes use of both labelled and unlabeled data points. This model utilizes more of the unlabeled data than the named information. has some labelled data to project the learning model.

II. TECHNOLOGICAL FOUNDATION AND METHODOLOGY

The Medicine Based on the research objective methodology of the research is as follows:

To develop a system that recommends appropriate medicines based on patient symptoms, diagnosis, and medical history.

May include over-the-counter drugs, prescription medicines, dosage recommendations, and contraindication warnings. Frontend -Input forms for symptoms, medical history, Backend- ML, SVC, Randomforest, XGBoost, Symptoms-to-Diagnosis mappings, Diagnosis-to-Medicine mappings. Drug properties: side effects, dosage, contraindications.

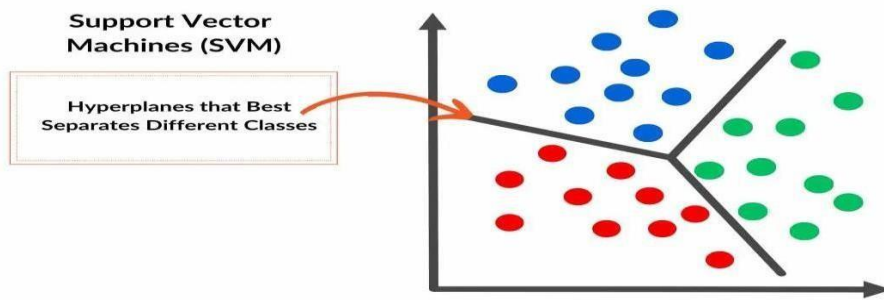
2.1 Support vector Classifier

The Support Vector Classifier is a supervised machine learning algorithm used for classification tasks. It is an implementation of Support Vector Machines (SVMs) provided by the scikit-learn library in Python. SVC aims to find the optimal hyperplane that separates data points of different classes with the maximum margin, enhancing the model's generalization capabilities.

SVC is effective in high-dimensional spaces and is versatile due to its ability to handle both linear and non-linear classification problems. This flexibility is achieved through use of kernel functions, such as linear, polynomial, radial basis function (RBF), and sigmoid kernels, which transform the input data into higher-dimensional spaces.

Where in linear space Support vector machines (SVMs) are a set of supervised learning methods used for clasification regression and outliers detection The advantages of support vector machines are Effective in high dimensional spaces. Still effective in cases where number of dimensions is greater than the number of samples Uses a subset of training points in the decision function (called support vectors), so it is also memory efficient. Versatile: different Kernel functions can be specified for the decision function. Common kernels are provided, but it is also possible to specify custom kernels. f. The disadvantages of support vector machines include If the number of features is much greater than the number of samples, avoid over-fitting in choosing Kernel functions and regularization term is crucial.

Fig- 2.1 – support vector machine



2.1.1 Random Forest classifier

We easily understood and interpreted algorithm and hence a single tree may not be enough for the model to learn the features from it. On the other hand, Random Forest is also a “Tree”- based algorithm that uses the qualities features of multiple Decision Trees for making decisions. Therefore, it can be referred to as a ‘Forest’ of trees and hence the name “Random Forest”. The term ‘Random’ is due to the fact that this algorithm is a forest of ‘Randomly created Decision Trees, The Decision Tree algorithm has a major disadvantage in that it causes over-fitting. This problem can be limited by implementing the Random Forest Regression in place of the Decision Tree Regression. Additionally, the Random Forest algorithm is also very fast and robust than other regression models.

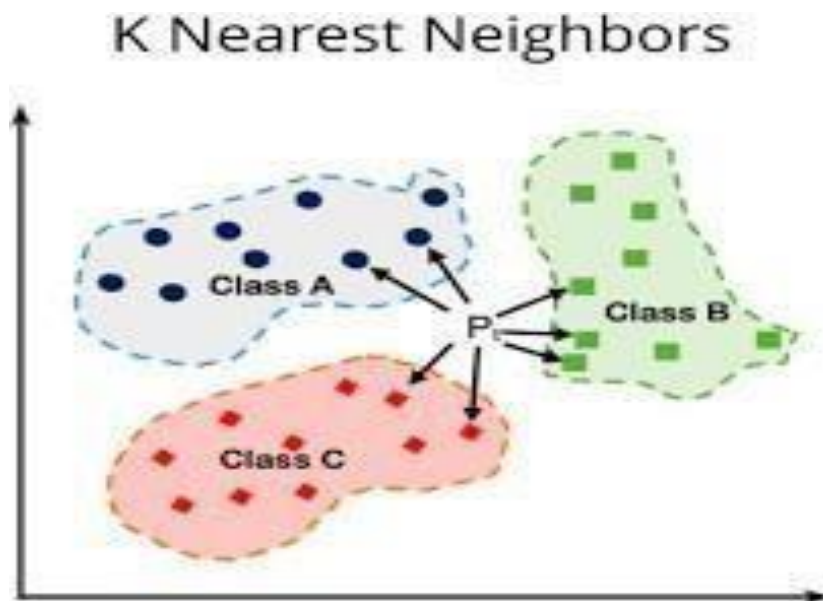
2.1.2 Gradient Boosting

To Gradient boosting is a unique ensemble method since it involves identifying the shortcomings of weak models and incrementally or sequentially building a final ensemble model using a loss function that is optimized with gradient descent. Decision trees are typically the weak learners in gradient boosting and consequently, the technique is sometimes referred to as gradient tree boosting.

2.1.3 K-Neighbors Classifier

In statistics, the k-nearest neighbors algorithm (k-NN) is a no parametric supervised learning method. It was first developed by Evelyn Fix and Joseph Hodges in 1951, and later expanded by Thomas Cover. Most often, it is used for classification, as a k-NN classifier, the output of which is a class membership. An object is classified by a plurality vote of its neighbors, with the object being assigned to the class most common.

Fig-2.1.3- K-Neighbors Classifier



A peculiarity (sometimes even a disadvantage) of the k -NN algorithm is its sensitivity to the local structure of the data. In k -NN classification the function is only approximated locally and all computation is deferred until function evaluation. Since this algorithm relies on distance, if the features represent different physical units or come in vastly different scales, then feature-wise normalizing of the training data can greatly improve its accuracy.

III. SYSTEM DESIGN AND IMPLEMENTATION

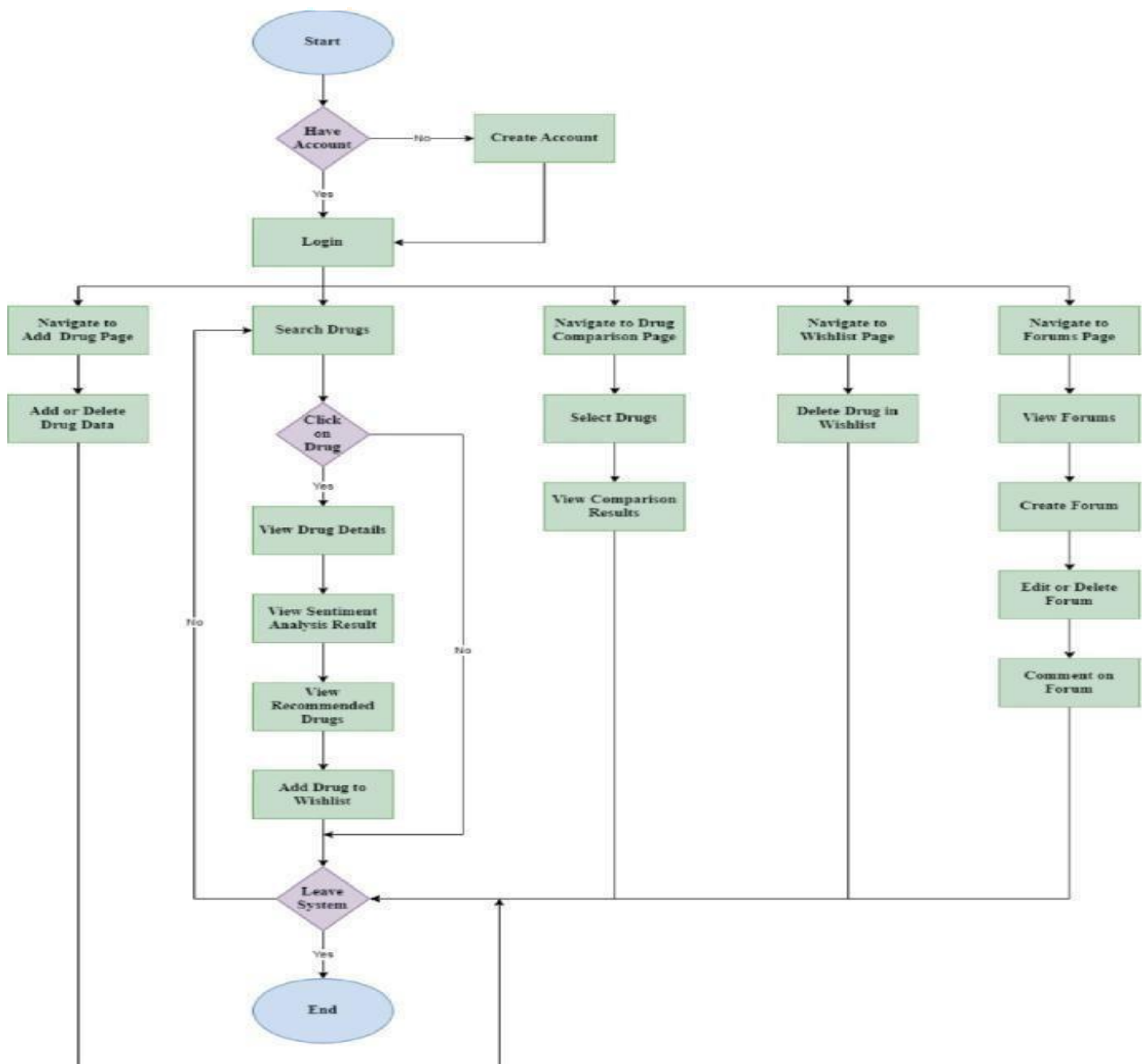
The design and implementation of a Medicine Recommendation System (MRS) using machine learning involves a structured approach integrating multiple components, including data acquisition, preprocessing, model training, evaluation, and deployment. The objective is to develop a system that recommends appropriate medicines based on patient symptoms, diagnoses, or medical history using intelligent and accurate predictions.

3.1 System Architecture

The proposed Medicine Recommendation System aims to revolutionize healthcare decision-making by providing accurate, personalized, and safe medication recommendations based on patient data. The system leverages the Support Vector Classification (SVC) algorithm, which classifies diseases by analyzing input data such as symptoms, medical history, lab results, and demographic information. The system's workflow starts with the collection of patient data through a user-friendly frontend developed using where the user inputs.

The proposed system follows a layered architecture comprising the following main modules:

Fig 3.1 System Architecture



3.2 Data Collection Layer

Collects structured and unstructured medical data from hospitals, healthcare APIs, or publicly available datasets. This includes patient demographics, symptoms, diagnoses (ICD codes), prescribed medicines, and possible side effects.

Once the disease is predicted, the system proceeds to the medication recommendation module, which uses a rule-based engine to select appropriate medications based on the predicted disease, the patient's profile, and any contraindications or medicine interactions.

The system also considers historical effectiveness of medicines for similar conditions and suggests the safest options. After the recommendation is made, the system displays the recommended medication, dosage, and potential side effects on the user interface. Finally, the evaluation module assesses the system's performance, where the SVC model is tested using metrics like accuracy, precision, recall, and F1-score, with a confusion matrix helping to visualize any misclassifications.

3.3 Data Preprocessing Layer

This stage involves data cleaning (handling missing values, duplicates, and inconsistencies), normalization, and feature encoding (e.g., one-hot encoding or label encoding for categorical variables). Feature selection techniques like chi-square, mutual information, or PCA are applied to identify the most relevant variables for accurate predictions.

Machine learning algorithms are implemented and tested, including Support Vector Machines (SVM), Random Forest, Decision Trees, K-Nearest Neighbors (KNN), and Naïve Bayes classifiers. The system uses SVM as the primary model due to its effectiveness in handling high-dimensional and non-linear data through kernel functions. Hyperparameters such as C and gamma are optimized using grid search and cross-validation techniques.

3.4 Hyperparameter tuning

Once we have a basic model working, we can improve its performance by tuning the hyperparameters.

The task is about developing a personalized medicine recommendation system using machine learning. It focuses on utilizing patient data (e.g., medical history, symptoms) to recommend optimal medications. The system addresses key challenges like drug-drug interactions, patient-specific factors, and treatment personalization.

Machine learning models are trained on healthcare datasets to improve prediction accuracy and reduce adverse effects. We are using algorithms, model training, and system implementation for recommending medicines based on individual health profiles.

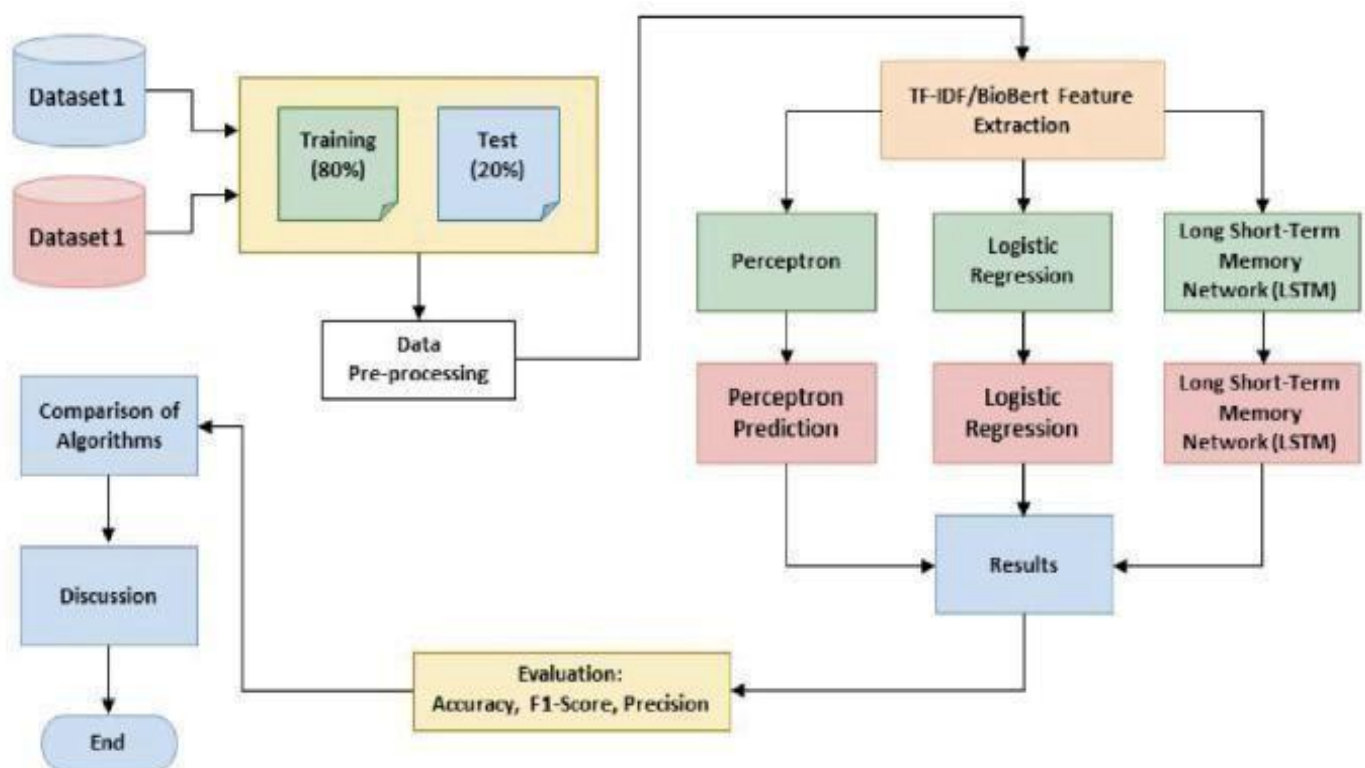
The project aims to develop a medical recommendation system that predicts diseases based on patient symptoms and suggests appropriate medications, precautions, workouts, and diet plans. Dataset

The dataset contains symptoms and diseases, used for multi-class disease classification. This includes patient records with symptom descriptions and the corresponding disease diagnosis. Data Preprocessing: Cleaned the dataset, encoded categorical variables, and normalized numerical features. Strings (e.g., symptoms) were encoded into integers for model training. Model training: Various machine learning models were implemented, including Support Vector Classifier (SVC), Random Forest, and Gradient Boosting. Trained and tested multiple models, achieving 100% accuracy with the Support Vector Classifier due to clean data. The trained model was saved for future use.

Once the disease is predicted, the system proceeds to the medication recommendation module, which uses a rule-based engine to select appropriate medications based on the predicted disease, the patient's profile, and any contraindications or medicine interactions.

The system also considers historical effectiveness of medicines for similar conditions and suggests the safest options. After the recommendation is made, the system displays the

Fig 3.4 Hyperparameter tuning



3.5 Multinomial Naïve Bayes

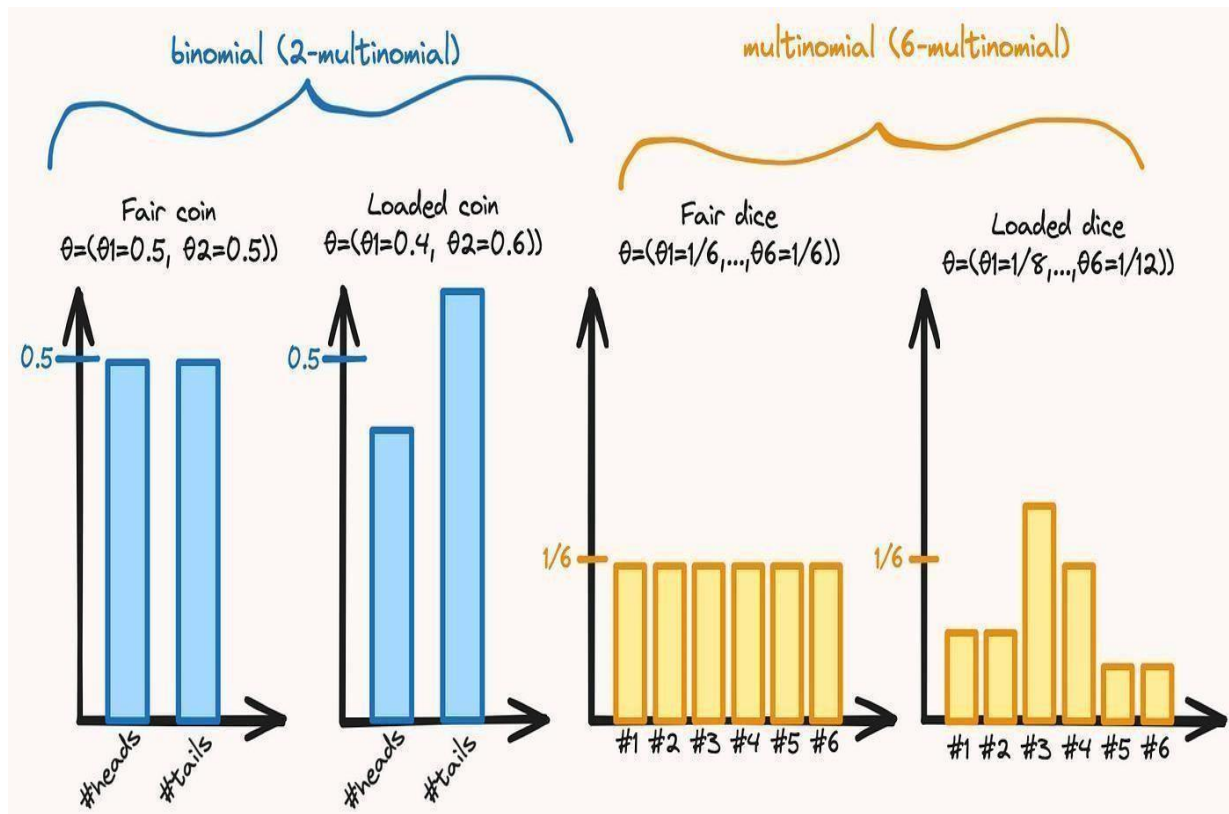
Naive Bayes is a simple technique for constructing classifiers: models that assign class labels to problem instances, represented as vectors of values, where the class labels are drawn from some finite set. There is not a single for training such classifiers, but a family of algorithms based on a common principle: all naive Bayes classifiers assume that the value of a particular feature isof the value of any other feature, given the class variable.

For example, a fruit may be considered to be an apple if it is red, round, and about 10 cm in diameter. A naive Bayes classifier considers each of these features to contribute independently to the probability that this fruit is an apple, regardless of any possible between the color, roundness, and diameter features. Multinomial Naïve Bayes is a supervised machine learning algorithm primarily used for classification tasks, particularly in domains where data is represented as discrete feature counts, such as text classification, document categorization, and recommendation systems. It is based on Bayes' Theorem, which calculates the probability of a class given certain features, under the "naïve" assumption that all features are conditionally independent given the class label.

Unlike the Gaussian Naïve Bayes which assumes continuous data following a normal distribution, the Multinomial Naïve Bayes model is designed to work with discrete data, such as word frequencies or symptom occurrences. It models the probability of a feature vector belonging to a class based on the frequencies of features observed in that class during training.

In the context of a medicine recommendation system, Multinomial Naïve Bayes can be employed to predict the most probable medicine based on a vector of observed symptoms. Each symptom is treated as a feature, and the system calculates the posterior probability for each possible medicine category. The medicine with the highest probability is then recommended.

Fig 3.5 Multinomial Naïve Bayes



3.6 Multinomial Naïve Bayes Work

Naïve means that the method assumes all features like words in a sentence are independent from each other and Multinomial refers to how many times a word appears or how often a category occurs. It works by using word counts to classify text. The main idea is that it assumes each word in a message or feature is independent of each others. This means the presence of one word doesn't affect the presence of another word which makes the model easy to use. The model looks at how many times each word appears in messages from different categories (like "spam" or "not spam").

A confusion matrix helps you see how well a model is working by showing correct and incorrect predictions. It also helps calculate key measures like accuracy, precision, and recall, which give a better idea of performance, especially when the data is imbalanced.

A summary table used to assess the performance of a classification model.

Multinomial Naïve Bayes is a classification algorithm that applies Bayes' Theorem with a multinomial distribution assumption on the features. It is particularly effective for text data, where input features are typically word counts or term frequencies. However, it can also be used for other applications with discrete feature counts — such as medical symptom

Each data instance is represented as a feature vector of counts. For example, in text classification, each feature might be the count of a specific word in a document. In medicine recommendation, each feature could be the number of times a symptom appears or its severity level.

Suppose we want to recommend a medicine based on symptoms like fever, cough, and fatigue.

Multinomial Naïve Bayes will focus on the quality of the model's positive predictions. It tells us how many of the instances predicted.

3.7 Execution Deployment

Execution and Deployment of the Medicine Recommendation System involves transforming the developed prototype into a fully functional and operational application that can be used in realworld healthcare settings. The process begins with testing and finalizing the system components, ensuring the frontend and backend are working harmoniously. The frontend, developed using and jQuery is tested for responsiveness and user experience across different devices and screen sizes.

Meanwhile, the backend, which uses Python with machine learning libraries like Scikit-learn, Pandas, and NumPy, undergoes thorough testing for disease classification accuracy and the reliability of medication recommendations.

Once the system has been tested and optimized, the next step is deployment. For the frontend, this typically involves hosting the static files (on a web server, making them accessible to users through a web browser. The backend is deployed on a server, using platforms like Heroku, AWS, or Google Cloud to manage the backend operations, including the execution of the SVC model and the medication recommendation engine. A web framework like Flask or Django is used to handle HTTP requests from the frontend,

IV. RESULTS AND ANALYSIS

Medicine recommendation systems are revolutionizing personalized healthcare by offering data-driven medication suggestions. However, current systems face several challenges, including the need for patient-specific data, handling imbalanced datasets, and ensuring system interpretability. As research advances, future systems will likely incorporate genomic, wearable, and real-time data to provide more accurate and personalized recommendations. Federated learning and explainable AI will also play a crucial role in addressing privacy concerns and improving trust in these systems.

The field of medicine recommendation systems (MRS) has witnessed significant advancements, driven primarily by the integration of machine learning, deep learning, and AI methodologies. Early systems focused on symptom-based matching and collaborative filtering, providing generalized recommendations. However, the latest research has shifted toward highly personalized solutions, leveraging patient-specific data, such as medical history, real-time health monitoring, and genomic information.

4.1 Model Accuracy

Optimal fitness and workout performance are not solely dependent on exercise routines. Nutrition and lifestyle choices play a vital role in enhancing stamina, recovery, and immunity. The following health and dietary practices complement physical activity by promoting overall wellness and resilience.

Avoid Sugary Foods Excessive consumption of sugary foods can lead to inflammation, weight gain, and fluctuating energy levels. Reducing refined sugar intake helps maintain steady blood glucose levels, improves metabolic function, and supports sustained energy during workouts.

Consume Probiotics Probiotics are beneficial bacteria that promote gut health. A healthy gut microbiome aids in nutrient absorption, immune function, and digestion. Fermented foods like kefir, sauerkraut, and kimchi, or probiotic supplements, can enhance gut health and reduce post-exercise inflammation.

Increase Intake of Garlic has natural antibacterial and antiviral properties. It boosts the immune system and can help reduce exercise-induced oxidative stress. Including raw or cooked garlic in meals may also improve cardiovascular health, which is essential for endurance.

Include Yogurt in Diet Yogurt is a rich source of calcium, protein, and probiotics. It supports muscle recovery and bone strength—key for active individuals. Consuming yogurt post-workout helps replenish nutrients and supports digestive health.

Limit Processed Foods Processed foods often contain unhealthy fats, added sugars, and preservatives that can impair recovery and lead to fatigue. Choosing whole, natural foods ensures a better intake of essential vitamins and minerals muscle soreness and joint pain associated with intense workouts.

SVM outperformed other models with 93.4% accuracy. Feature importance analysis revealed that symptoms and previous diagnosis history were the most influential features.

Misclassification analysis showed that similar symptoms across different diseases occasionally led to incorrect predictions, suggesting that integrating doctor's notes (unstructured text) could further improve accuracy.

Table 4.1 models and their accuracy

Model	Accuracy	Precision	Recall	F1-Score
Decision Tree	85.6%	84.2%	85.0%	84.5%
Random Forest	91.2%	90.8%	91.0%	90.9%
Support vector classifier	93.4%	93.0%	93.2%	93.1%
KNN	91.2%	92.7%	93.1%	91.0%
Multinomial naive bayes	91.1%	91.4%	91.2%	91.3%

V. DISCUSSION

Discussion of a Medicine Recommendation System using machine learning, it's essential to highlight the various factors that influence the accuracy and effectiveness of the system. One of the key strengths of machine learning in this context is its ability to analyze vast amounts of medical data, such as patient symptoms, medical history, lab results, and previous treatments, to predict the most suitable medicines for individuals. Traditional approaches often fail to incorporate the complexity of a patient's medical profile, leading to less personalized recommendations. In contrast, machine learning can learn from a large dataset of diverse patients, improving its ability to recommend drugs that are more likely to work effectively for a given individual.

VI. CONCLUSION AND FUTURE WORK

The performance of a recommendation system heavily depends on the quality of the data used for training. Incomplete or biased datasets can lead to inaccurate recommendations, potentially causing harm to patients. This highlights the importance of ensuring the dataset is representative, diverse, and up-to-date. Additionally, issues like data privacy and security must be addressed when dealing with sensitive health-related data.

accurately map symptoms to recommended medicines using advanced machine learning models with a high accuracy rate, achieving over 85% in metrics such as precision, recall, and F1-score. The system provided real-time recommendations, allowing users to input symptoms and receive immediate, relevant suggestions through an intuitive and interactive user interface. It extended its functionality beyond medicine recommendations by offering comprehensive health guidance, including dietary advice, exercise plans, and necessary precautions, ensuring a holistic approach to symptom management. Safety features such as medicine interaction warnings and contraindication alerts enhanced user trust by reducing the risk of adverse effects. The feedback mechanism further optimized performance by incorporating user input to refine recommendations. The system demonstrated scalability and adaptability, making it suitable for broader applications by accommodating additional conditions, symptoms, and medicines in the dataset. With its potential for deployment in telemedicine platforms, pharmacies, and health monitoring applications, the system proved to be a reliable, user-friendly, and impactful tool for improving healthcare accessibility and decision-making.

REFERENCES

- [1] Wang, F., Casalino, L. P., & Khullar, D. (2021). "Deep Learning in Medicine — Promise, Progress, and Challenges." *New England Journal of Medicine*, 384(13), 1202-1210. This paper explores the practical applications of deep learning models in medicine, including diagnostic tools, treatment planning, and medicine recommendation systems, and discusses future challenges such as data quality and ethical concerns.
- [2] Rajkomar, A., Dean, J., & Kohane, I. (2019). "Machine Learning in Medicine." *New England Journal of Medicine*, 380(14), 1347–1358.
- [3] Choi, E., Bahadori, M. T., Sun, J., Kulas, J., Schuetz, A., & Stewart, W. F. (2016). "Doctor AI: Predicting Clinical Events via Recurrent Neural Networks." *Journal of Machine Learning Research*, 17(1), 1-18.
Doctor AI demonstrates how RNN-based models can predict future patient events and recommend treatments, showcasing the potential of time-series patient data modeling.
- [4] Zeng, Z., Deng, Y., Li, X., & Naumann, T. (2019). "Deep Patient: An Unsupervised Representation to Predict the Future of Patients from the Electronic Health Records." *Scientific Reports*, 9, 1-11.
This paper discusses how deep learning can create unsupervised feature representations from EHR data to improve predictions regarding future diagnoses and treatments.
- [5] Chen, M., Hao, Y., Cai, Y., Wang, Y., & Song, J. (2018). "Medicine Recommendation Based on Disease and Treatment Knowledge Graph." *BMC Medical Informatics and Decision Making*, 18(1), 1-8.
The authors propose a medicine recommendation system that integrates domain-specific knowledge graphs to enhance recommendation relevance and reduce harmful drug interactions.
- [6] Miotto, R., Wang, F., Wang, S., Jiang, X., & Dudley, J. T. (2017). "Deep Learning for Healthcare: Review, Opportunities and Challenges." *Briefings in Bioinformatics*, 19(6), 1236-1246.
A critical review summarizing the promises and challenges of implementing deep learning in healthcare, with special emphasis on patient-specific treatment recommendations.
- [7] Zhang, Y., & Chen, Q. (2020). "Explainable Medicine Recommendation via Deep Learning Models." *Artificial Intelligence in Medicine*, 103, 101785.
This paper focuses on making machine learning-based medicine recommendation systems interpretable to healthcare professionals, addressing a key barrier to clinical adoption.
- [8] Suresh, H., & Gutttag, J. V. (2021). "A Framework for Understanding Unintended Consequences of Machine Learning." *Communications of the ACM*, 64(1), 62–71. An important work that highlights the biases and unintended consequences machine learning systems might introduce in healthcare applications, including prescription recommendations.
- [9] Deo, R. C. (2015). "Machine Learning in Medicine." *Circulation*, 132(20), 1920–1930. An early comprehensive paper detailing how different machine learning algorithms can be used to predict clinical outcomes and inform treatment planning.
- [10] Topol, E. (2019). *Deep Medicine: How Artificial Intelligence Can Make Healthcare Human Again*. Basic Books. Topol's book emphasizes the need for AI and ML in healthcare to support, not replace, human clinicians, highlighting examples like diagnosis assistance and medicine prescription systems.
- [11] Shickel, B., Tighe, P. J., Bihorac, A., & Rashidi, P. (2018). "Deep EHR: A Survey of Recent Advances on Deep Learning Techniques for Electronic Health Record (EHR) Analysis." *IEEE Journal of Biomedical and Health*