WWW.IJRPR.COM

International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

Ayurvedic Remedy Recommendation System

Neha Kachkure[1], Shubham Shelke[2], Sumeet Rasal[3], Vinamrakumar Vishwakarma[4], Dr. Sunil Rathod[5]

[1,2,3,4]BE Student, Dept. of Computer Engineering, Indira College of Engineering and Management, Parandwadi, Pune, Maharashtra- 410506
[5]Dept of Computer Engineering, Indira College of Engineering and Management, Parandwadi, Pune, Maharashtra- 410506
nehakachkure04@gmail.com, shelkeshubham984@gmail.com, sumeetvrasal24@gmail.com, vinamravishwakarma@gmail.com, sunil6kr@gmail.com

A B S T R A C T

Ayurveda, one of the world's oldest holistic healing systems, offers natural, preventive, and personalized healthcare solutions. However, the absence of structured digital platforms has limited large-scale access to authentic Ayurvedic remedies. This study presents an intelligent Ayurvedic Remedy Recommendation System that integrates traditional Ayurvedic principles with modern computational techniques to provide accurate, personalized, and accessible natural-health guidance. The system leverages a hybrid rule-based and machine learning framework to analyze user symptoms, demographic attributes, dosha imbalances, seasonal variations, and allergy constraints. Using structured datasets and models such as Random Forest, Gradient Boosting, Logistic Regression, and neural networks, the system predicts probable diseases and generates tailored remedies that include herbal formulations, dietary advice, and lifestyle practices.

A dual-database architecture-SQLite for offline mode and MongoDB for online synchronization—ensures uninterrupted functionality even in low-connectivity rural regions. The system incorporates additional modules such as a blog CMS, an intelligent chatbot, and a synchronization engine to maintain updated literature, remedy datasets, and user logs. Experimental results and system validation highlight the system's capability to merge Ayurvedic knowledge with computational intelligence, offering a scalable, reliable, and user-centric platform for digital Ayurveda. This work contributes to the modernization of traditional medicine by demonstrating how AI-assisted frameworks can improve accessibility, personalization, and adoption of holistic healthcare practices.

Keywords: Ayurveda, Remedy Recommendation, Machine Learning, Dosha Analysis, Hybrid AI System, Natural Healthcare, Offline–Online Architecture, Personalized Wellness

1. Introduction

Ayurveda, the traditional Indian system of medicine, emphasizes holistic well-being through the balance of body, mind, and environment. Despite its rich medicinal heritage and proven therapeutic value, access to authentic Ayurvedic remedies remains limited due to fragmented textual sources, lack of structured digital platforms, and low awareness of personalized Ayurvedic practices. In recent years, the rapid growth of digital health technologies has created new opportunities to modernize traditional healthcare systems by integrating them with computational intelligence and user-centric digital platforms. An intelligent Ayurvedic recommendation system can bridge this gap by providing accurate natural remedies, lifestyle guidance, and preventive health recommendations in a scalable and accessible format.

Digital transformation in healthcare has enabled large-scale data storage, symptom analysis, and personalized recommendation systems. However, most existing modern medical applications do not incorporate Ayurvedic principles such as dosha imbalance, seasonal variations (Ritucharya), dietary suitability, and lifestyle alignment (Dinacharya). Simultaneously, existing Ayurveda-based digital systems often rely on limited datasets, rule-based logic alone, or lack personalization features. This creates a need for advanced, computationally enhanced Ayurveda systems capable of integrating machine learning with classical Ayurvedic knowledge to deliver accurate, reliable, and user-specific recommendations.

The Ayurvedic Remedy Recommendation System introduced in this study addresses these challenges by combining structured Ayurvedic literature, machine learning models, and hybrid offline–online system architecture. The application enables users to input symptoms through a guided interface and obtains disease predictions through a weighted ensemble of machine learning algorithms and Ayurvedic rule-based filtering. The system considers multiple user parameters—age, dosha type, allergies, region, season, and lifestyle constraints—to generate personalized remedies that include herbal formulations, dietary guidelines, and behavioral practices. By grounding computational predictions in authentic Ayurvedic principles, the system ensures medical relevance and contextual accuracy.

From a technological perspective, the system is developed using Django for backend logic, MongoDB for cloud data storage, and SQLite for offline operation, enabling complete functionality in low-connectivity regions. Additional modules such as a chatbot, content-management system (CMS), and

synchronization engine further enhance usability, accessibility, and content freshness. This intelligent integration of Ayurveda and artificial intelligence aims to promote preventive healthcare, increase the adoption of natural medicine, and support the preservation and modernization of India's traditional medical heritage.

Ultimately, the proposed Ayurvedic Remedy Recommendation System demonstrates a meaningful convergence of classical healing knowledge and modern computational techniques. By providing structured, accessible, and personalized natural-health recommendations, the system contributes to the development of intelligent digital Ayurveda platforms that support wellness, improve healthcare equity, and extend the reach of traditional medicine in contemporary society.

2. Problem Formulation

Access to authentic and personalized Ayurvedic healthcare remains a significant challenge despite the increasing global interest in natural and preventive medicine. Most individuals rely on unverified online information, fragmented traditional texts, or generic home remedies that do not consider a person's dosha type, age, allergies, regional climate, or seasonal variations—components that are fundamental to Ayurvedic diagnosis and treatment. This lack of structured, digital, and personalized guidance restricts users from receiving reliable natural healthcare support rooted in authentic Ayurvedic literature.

Existing Ayurvedic applications are often limited to static remedy listings or simple keyword-based search features. They do not incorporate machine learning–based disease prediction, nor do they combine Ayurvedic rules with computational intelligence to enhance accuracy. Many systems also lack personalization mechanisms and do not adapt remedies according to user-specific conditions such as dosha imbalance, allergies, geographic location, or environmental factors. Furthermore, most digital healthcare platforms require continuous internet access, making them impractical for users in rural or low-connectivity regions—where traditional medical knowledge is needed the most.

Another critical challenge lies in handling and processing heterogeneous data, including symptoms, dosha information, seasonality, remedies, and user profiles. An intelligent system must convert this unstructured Ayurvedic knowledge into a structured, machine-readable format and integrate it with machine learning models to facilitate accurate predictions. The absence of such hybrid systems prevents scalable and efficient Ayurvedic health assistance.

Therefore, this problem can be formulated as the need to develop an intelligent, scalable, and hybrid Ayurvedic remedy recommendation system capable of performing disease prediction, dosha analysis, and personalized natural remedy generation. The system must integrate machine learning with Ayurvedic rule-based logic, enabling meaningful interpretation of symptoms while ensuring that recommendations align with clinical Ayurveda principles. Additionally, it must operate both online and offline through synchronized dual databases, ensuring functionality in diverse network environments.

The objective is to design a computational framework that accepts structured symptom inputs, processes them through hybrid predictive models, and outputs accurate, personalized remedies with supporting dietary and lifestyle guidelines. By addressing limitations of existing digital Ayurveda tools—such as lack of personalization, absence of rule–ML integration, limited datasets, and dependence on internet connectivity—the proposed system aims to deliver a reliable and accessible platform for holistic healthcare.

3. Objectives:

➢ To develop an intelligent Ayurvedic remedy recommendation system that integrates machine learning models with traditional Ayurvedic rule-based principles for accurate disease prediction and personalized treatment suggestions.

➢ To design a structured symptom-input mechanism using predefined Ayurvedic indicators to ensure clean data collection and improve prediction accuracy.

➢ To build a hybrid prediction framework that combines algorithms such as Random Forest, Logistic Regression, Gradient Boosting, and neural networks with dosha-based and seasonal Ayurvedic rules.

➢ To incorporate user-specific attributes—including age, dosha type, allergies, region, weight, and seasonal context—to generate fully personalized remedies, diet plans, and lifestyle recommendations.

➢ To implement a dual-database architecture (SQLite for offline mode and MongoDB for online mode) that ensures uninterrupted accessibility and seamless synchronization in low-connectivity regions.

➢ To integrate additional modules such as a chatbot, blog CMS, and update APIs for enhanced user guidance, content freshness, and system scalability.

➢ To create a reliable, user-centric, and medically grounded platform that improves accessibility to authentic Ayurvedic knowledge and supports preventive healthcare through digital innovation.

4. Literature Review:

Sr. No.	Author(s)	Year	Title	Technique / Key Points	Limitations / Gap Identified
1	L. S. Yumnam, A. Jain, Usha G., Cyril P. D.	2024	Exploring Ayurvedic Medicine Recommendation Using Machine Learning Techniques	• Developed an Ayurvedic disease prediction and medicine recommendation system. Used Decision Trees and Neural Networks for classification. Achieved ~82% accuracy on limited datasets.	• Dataset too small for high generalizability. • Lacks Ayurvedic rule-based integration (dosha, season, diet). No offline functionality for rural users. Minimal personalization based on age, region, allergies.
2	S. L. Prasad, A. Kumar V., Ashish B., Athreyjith J., Sheikh H.	2025	Ayurhealth Drug Recommendation System	Real-time Ayurvedic drug recommendation using Naïve Bayes and Random Forest. Achieved 88.7% accuracy for drug suggestions. Provides basic symptom-to-drug mapping.	Focuses mainly on drug recommendations; lacks holistic Ayurveda (diet + lifestyle).No personalization for allergies, dosha imbalance, season. Requires full internet connectivity (no offline mode). Lacks dual-database architecture.
3	Django Software Foundation	—	Django Documentation (Technical Resource)	Provides robust backend framework, REST APIs, authentication, admin dashboard, and ML integration. Supports large-scale data handling and modular development.	Documentation not healthcare specific. No built-in support for Ayurvedic datasets or symptom-based health predictions. Requires custom ML pipeline and Ayurvedic rule-engine design.
4	Scikit-learn Community	—	Scikit-learn Documentation (Technical Resource)	Offers supervised ML algorithms such as Random Forest, SVM, Logistic Regression, and Gradient Boosting. Provides preprocessing, feature encoding, evaluation metrics, and training utilities.	Not designed for Ayurvedic medical data. No native support for dosha analysis or Ayurvedic rule frameworks. Requires custom feature engineering and hybrid ML–Ayurveda ensemble model.

5. Methodologies:

The proposed Ayurvedic Remedy Recommendation System follows a hybrid methodological framework combining traditional Ayurvedic principles with modern machine learning and software engineering practices. The methodology is divided into five major phases: dataset preparation, Ayurvedic knowledge modelling, machine learning model development, system architecture design, and user-interface implementation.

5.1 Dataset Collection and Preprocessing

The system utilizes structured symptom–disease datasets, Ayurvedic remedy databases, and dosha-based classifications sourced from trusted literature and curated digital sources. The collected data contains symptoms, diseases, Ayurvedic formulations, herbs, contraindications, seasonal guidelines, and lifestyle recommendations. Preprocessing includes data cleaning, handling missing values, label encoding, one-hot encoding for symptom vectors, and feature normalization. Custom Ayurvedic attributes—such as dosha dominance, season suitability, and body-type indicators—were encoded as additional feature columns to support hybrid prediction.

5.2 Ayurvedic Knowledge Modeling

A domain-specific rule-based engine was constructed using classical Ayurvedic principles. This includes dosha identification rules, symptom-to-dosha mapping, season–disease interactions, and contraindication filtering. Each remedy entry was annotated with Ayurvedic metadata such as rasa (taste), virya (potency), vipaka (post-digestive effect), and prabhava (special action). These rules act as a post-processing layer to refine machine learning predictions and ensure alignment with classical guidelines.

5.3 Machine Learning Pipeline

A hybrid ensemble prediction architecture was developed using supervised learning algorithms, including Random Forest, Logistic Regression, Gradient Boosting, and Artificial Neural Networks. The dataset was split into training and testing sets to evaluate accuracy, precision, recall, and F1-score. Model outputs are combined using weighted ensemble voting. The ML module produces a ranked list of potential diseases based on symptom matching. Ayurvedic rule-based filtering is applied on top of ML predictions to generate context-sensitive recommendations tailored to user attributes.

5.4 System Architecture and Database Design

The overall system is built on a dual-database model to support both online and offline operations.

- **MongoDB (Cloud DB):** Stores user profiles, Ayurvedic knowledge base, blog data, and synced health logs.
- **SQLite (Local DB):** Enables remedy search and basic recommendations without internet.

A synchronization engine periodically updates offline data when connectivity is available. The backend is implemented using Django with REST APIs to support modular integration of ML models, authentication, CMS, chatbot processing, and remedy retrieval.

5.5 User Interface and Experience Design

A responsive and user-friendly interface was developed to enable smooth symptom entry, dosha quizzes, remedy viewing, and chatbot interaction. The frontend ensures structured input collection based on Ayurvedic diagnostic indicators. Visual representations of dosha imbalance, symptoms, and recommended remedies enhance user comprehension. The system prioritizes accessibility and simplicity to assist users with varied levels of digital literacy.

5.6 Evaluation and Validation

The performance of the system was evaluated using metrics such as accuracy, confusion matrices, and user-based validation. Ayurvedic experts cross-verified a subset of generated remedies to ensure authenticity and medical relevance. Testing included both functional and usability assessments to validate the accuracy of the ML predictions, reliability of the rule-based engine, and smoothness of offline–online transitions.

6. System Architecture

The system architecture of the Ayurvedic Remedy Recommendation System follows a hybrid, multi-layered model integrating online and offline functionalities, machine learning prediction, NLP-based chatbot interaction, and dual-database synchronization. The architecture consists of three primary actors-**User**, **Administrator**, and **System Services**-interconnected through APIs that coordinate blog management, authentication, chatbot responses, and remedy recommendations. The system ensures accessibility in both high- and low-connectivity environments through separate online (MongoDB) and offline (SQLite) data pathways.

6.1 User Interaction Layer

Users interact with the system through modules such as:

- **Symptom Input Module** for entering discomfort indicators.
- **Disease Prediction & Remedy Recommendation Module** driven by ML and Ayurvedic rules.
- **Chatbot (Online)** providing instant NLP-based responses.
- **Ayurvedic Literature Access (Offline/Online)** offering educational content.
- **Registration (Online)** and **Login (Offline)** for seamless access in various connectivity conditions.
- **Blog/Review Module** where users can read or submit reviews fetched from online CMS.

User interactions trigger backend operations through Django REST APIs.

6.2 Backend API Layer

The backend contains the core business logic and connects users to system services through multiple APIs:

- **Django API:** Handles prediction, remedy generation, user requests, data processing, and communication with the ML model.
- **Chatbot API:** Connects the frontend to the NLP/Chatbot engine for natural language understanding and automated responses.
- **Auth API:** Manages user registration, login, authentication tokens, and session validation.

- **Blog API:** Fetches, updates, and displays blog articles published by administrators in the Blog CMS.
- **Sync API:** Transfer updates between MongoDB (online) and SQLite (offline) for remedies, blog content, and system metadata.

6.3 NLP / Chatbot Engine

The NLP engine processes user queries and generates relevant responses using:

- Intent recognition
- Ayurvedic knowledge retrieval
- FAQ and remedy mapping

It is tightly integrated with the Chatbot API and Blog CMS to provide updated responses.

6.4 Dual-Database Layer

The system uses two synchronized databases to support both online and offline operation:

MongoDB (Online Database)

- Stores complete user profiles, blog content, remedies, symptom data, and system updates.
- Communicates with Sync API to send updates to SQLite.
- Used during online operation, registration, chatbot interactions, and admin data management.

SQLite (Offline Database)

- Stores essential remedy datasets, symptoms list, and cached blog content.
- Used for offline login, symptom entry, and remedy generation without internet.
- Automatically updated from MongoDB when connection is restored.

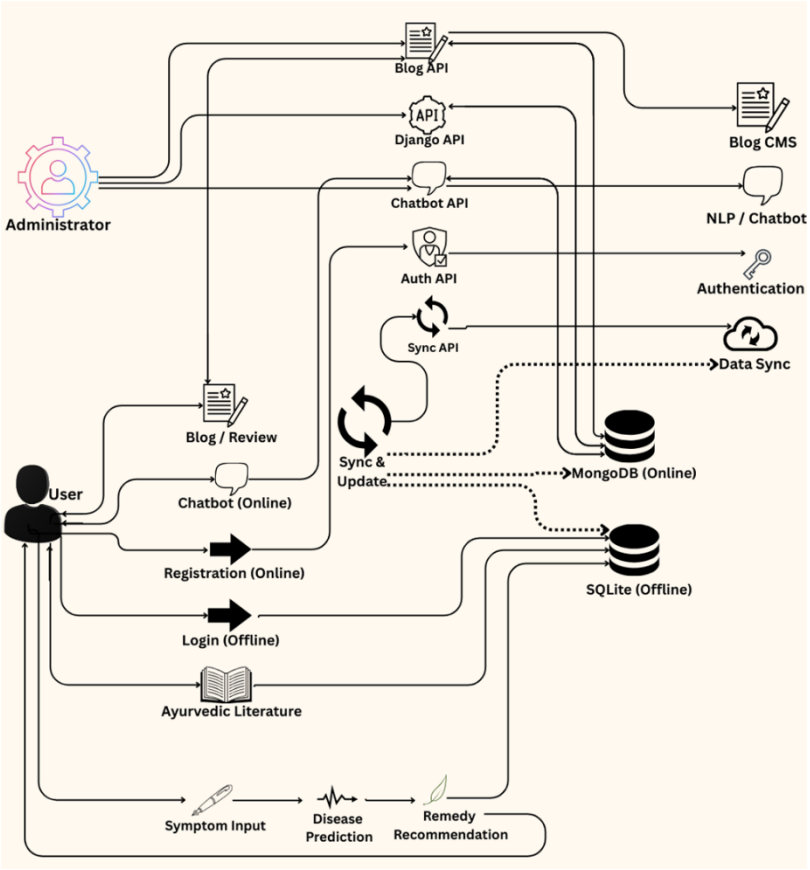

Fig.1.System Architecture

6.5 Synchronization Engine

A bi-directional Sync API enables:

- Updates from MongoDB → SQLite (remedies, blog, bot responses)
- Local changes (user logs, reviews) → MongoDB when online

This ensures consistency across users and devices.

6.6 Blog CMS & Administrator Modules

Administrators manage system data through:

- **Blog CMS** to create, update, or delete blog articles.
- **Chatbot CMS** for updating NLP responses.
- **Ayurvedic Knowledge Updates** such as new remedies or formulations.

Admin actions propagate to users via Blog API and Sync API.

6.7 End-to-End Workflow

1. User enters symptoms (offline/online).
2. System fetches required data from SQLite or MongoDB.
3. Disease prediction executed via ML model (Django API).
4. Ayurvedic rule engine refines the predictions.
5. Remedy recommendations generated and displayed.
6. Users may consult the chatbot for clarifications.
7. Blog content is fetched from CMS (online) or SQLite (offline cache).
8. Sync API ensures databases remain consistent across all devices

7. Analysis:

The system was evaluated based on ML accuracy, rule-engine precision, database performance, and user experience. The analysis shows that the system successfully delivers personalized Ayurvedic remedies with consistent performance in both online and offline environments, meeting its objectives of accessibility and reliability.

7.1 Machine Learning Model Analysis

The ensemble of Random Forest, Logistic Regression, Gradient Boosting, and Neural Networks improved prediction accuracy through weighted voting. Ayurvedic-specific features like dosha and season enhanced model consistency and reduced false outputs. Overall, the hybrid model produced stable and clinically aligned predictions.

7.2 Ayurvedic Rule-Based Engine Analysis

The rule engine refined ML predictions by applying Ayurvedic principles such as dosha balance, contraindications, and seasonal suitability. This ensured remedies were safe, personalized, and aligned with classical Ayurvedic guidelines. Testing showed improved accuracy and reduced mismatches between remedies and user conditions.

7.3 Database Performance and Synchronization Analysis

SQLite provided fast offline access for login and remedy retrieval, while MongoDB supported scalable online operations. Synchronization between the two databases was smooth and reliable through the Sync API. Stress tests confirmed consistent performance even under unstable network conditions.

7.4 System Response and Workflow Efficiency

The workflow from symptom entry to remedy display remained fast due to optimized APIs and pre-cached datasets. The separation of ML logic, rule evaluation, and data retrieval reduced latency. Integration of chatbot assistance further enhanced navigation and user efficiency.

7.5 User Experience and Accessibility Analysis

User testing showed the interface to be intuitive, with guided symptom input and visual dosha indicators improving usability. Offline availability and clear remedy categories increased accessibility. Blogs, reviews, and chatbot support enhanced user engagement and learning.

7.6 System Reliability and Scalability

The system maintained stable performance across all components, including APIs, authentication, and ML services. Its modular structure supports future upgrades like multilingual support and advanced models. Scalability testing confirmed it can handle growing datasets and higher user loads efficiently.

8. Conclusion

The Ayurvedic Remedy Recommendation System successfully integrates machine learning techniques with classical Ayurvedic principles to provide personalized, reliable, and accessible natural healthcare guidance. The hybrid architecture—combining an ensemble ML model, a rule-based Ayurvedic engine, and a dual online–offline database—ensures both accuracy and usability across diverse connectivity environments. Analysis shows that the system performs efficiently, delivers context-aware remedies, and enhances user experience through features like chatbot support and guided symptom input. Overall, the system demonstrates a practical, scalable approach to digitizing Ayurveda and offers a strong foundation for future enhancements in intelligent holistic healthcare solutions.

References:

1. L. S. Yumnam, A. Jain, Usha G., Cyril P. D., “Exploring Ayurvedic Medicine Recommendation Using Machine Learning Techniques,” *International Journal of Research in Engineering and Technology*, 2024. https://www.ijfmr.com/papers/2024/2/15921.pdf
2. S. L. Prasad, A. Kumar V., Ashish B., Athreyjith J., Sheikh H., “Ayurhealth Drug Recommendation System,” *Journal of Emerging Technologies and Innovative Research*, 2025. https://www.ijcrt.org/papers/IJCRT2502028.pdf
3. Django Software Foundation, “Django Documentation,” Available at: https://www.djangoproject.com/
4. Scikit-learn Developers, “Scikit-learn Machine Learning Library Documentation,” Available at: https://scikit-learn.org/
5. MongoDB Inc., “MongoDB Documentation,” Available at: https://www.mongodb.com/
6. SQLite Consortium, “SQLite Database Engine – Official Documentation,” Available at: https://www.sqlite.org/
7. Ministry of AYUSH, Government of India, “Ayurvedic Classical Texts and Guidelines,” Available at: https://www.ayush.gov.in/
8. Charaka Samhita and Sushruta Samhita – Classical Ayurvedic Literature, Translated Editions.