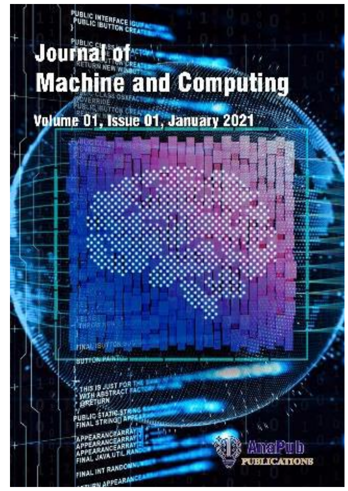
# Journal Pre-proof

A Robust Deep Learning Computational Model to Provide Recommendation for Healthcare Support using Segmentation Methodology




**Madula C and Vignesh U**

DOI: 10.53759/7669/jmc202505147

Reference: JMC202505147

Journal: Journal of Machine and Computing.

Received 24 January 2025

Revised from 06 April 2025

Accepted 18 June 2025

**Please cite this article as:** Madula C and Vignesh U, “A Robust Deep Learning Computational Model to Provide Recommendation for Healthcare Support using Segmentation Methodology”, Journal of Machine and Computing. (2025). Doi: https:// doi.org/10.53759/7669/jmc202505147.

# A Robust Deep Learning Computational Model to Provide Recommendation for Healthcare Support using Segmentation Methodology

C. Madula, U. Vignesh*
School of Computer Science and Engineering, Vellore Institute of Technology, Chennai
*vignesh.u@vit.ac.in
*Corresponding Author: U.Vignesh

**Abstract**

The recent revival in the popularity of Ayurvedic medicine demands the smart digital systems which will be able to prescribe medicinal herbs according to the individualized symptom picture. In this paper, the proposed method provides a lightweight and explainable hybrid model, termed as MedLeafRec, which can suggest the Ayurvedic medicinal leaves and their suitable dosage, given the input features, i.e., age, gender, type of symptom, temperature, and severity. MedLeafRec incorporates a two-level decision making method: rule-based inference engine, which relies on Ayurvedic expertise, and a fallback decision tree classifier, which deals with the situations that were not covered by predefined mappings. Prediction of dosage is achieved by using a linear regression model that incorporates the use of normalized physiological parameters to predict quantity in either grams or milliliters. Comprehensive testing on a selected dataset proves that MedLeafRec has a dosage prediction Mean Absolute Error (MAE) of 0.62 g/ml and a classification accuracy of 95.34%. Such performances are substantially higher than those of baseline models, such as Random Forest (89.45%), SVM (87.50%), and Rule-Only Systems (82.35%). In addition, the model has a small footprint (2.1 MB) and low inference latency (3.4 ms/sample), which makes it very applicable in mobile and constrained settings. The modular and transparent design of MedLeafRec allows it to integrate with healthcare platforms that can be deployed in the field without disturbing the clinical reasoning of the conventional practice.

**Keywords:** Ayurveda, Medicinal Leaf Recommendation, Rule-Based Reasoning, Decision Tree Classifier, Dosage Prediction, Herbal Medicine, Linear Regression, interpretable AI.

## 1. Introduction

Ayurveda is considered one of the most ancient systems of holistic medicine, which is still centrally featured in health and wellness ecosystems in South Asia and other regions. As the world moves toward personalized, natural and preventive health care, the need of the hour is intelligent systems which are capable of mining the Ayurvedic wisdom and applying it to contemporary uses is increasing at an unprecedented rate [1] [2]. Ayurvedic pharmacopoeia includes herbal leaf-based preparations like Tulsi in fever and Giloy in immunity-boosting preparations. Nevertheless, the diagnosis process and choosing the appropriate leaf and dosage by the traditional practitioners is rather subjective, region-specific and cannot be easily scaled. Accordingly, there is a pressing necessity of computational models that may decode symptomatic patterns and prescribe the corresponding medicinal leaves and dosage in a manner that is both evidence-based and adheres to Ayurvedic ideologies [3] [4] [5].

Machine learning and artificial intelligence have evolved and changed healthcare diagnostics to a considerable extent. Support Vector Machines (SVM), Decision Trees, Random Forests, and Logistic Regression models have performed predictive disease and drug suggestion tasks quite extensively in the allopathic realms [6] [7]. In more recent years, more complicated deep learning models such as Convolutional Neural Networks (CNNs) and Long Short-Term Memory (LSTM) networks have demonstrated potential in learning intricate patterns in medical data. Nevertheless, these models fail on two counts, interpretability and domain alignment, when considered in the setting of Ayurvedic recommendation systems. Although deep models have the potential to achieve high accuracy, they are usually viewed as black boxes and their predictions cannot be explained easily to make them trustworthy to the practitioners or the users [8] [9] [10]. Moreover, the majority of current models have not been adapted to the peculiarities of logic and combinatorial symptomatology interpretation that is used in Ayurveda, in which the scenarios of symptom-remedy correlations lie within centuries of traditional expertise, but not merely in statistical co-occurrence.

Past work on automating Ayurvedic recommendation systems has been based on rule-based engine only or direct machine learning classifier. Rule-based systems represent Ayurvedic logic in hard coded rules, e.g. “when fever and cough then suggest Tulsi.” Though these systems are explainable and grounded in domain knowledge, they are brittle and have poor coverage [11] [12] [13]. Unknown or vague combinations of symptoms usually produce inconclusive results. Conversely, the purely statistical classifiers have the ability to generalize more, but they eliminate the conventional knowledge representation and are less interpretable. These systems can suggest a solution that does not correspond to Ayurvedic reasoning, which damages credibility and cultural acceptability to the user. [14] [15] Figure 1 illustrates important Ayurvedic leaves, showcasing medicinal plants.

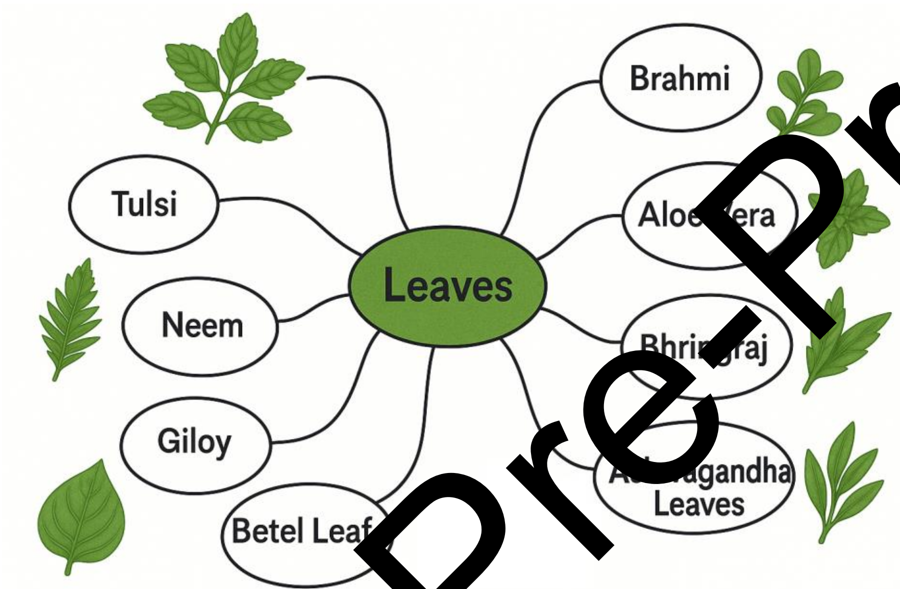



**Figure 1: Important Ayurvedic Leaves**

In order to alleviate these shortcomings, this paper introduces MedLeafRec, a unified framework that combines the better of the two paradigms. It employs a rule-augmented strategy in which a hand-crafted knowledge base of Ayurvedic rules can be utilized as the initial inference layer. In case the symptoms of a patient correspond to any of these high-confidence rules (empirically supported by training data), the respective medicinal leaf is directly suggested. This guarantees that system maintains the interpretability and classical faithfulness of the classical practice. Nevertheless, when the input data is not covered by these preimplemented rules or results in several contradictory outputs, a supervised machine learning model, a decision tree classifier, is called as a safeguard measure. Trained on encoded symptom vectors and patient metadata this model learns to generalize on historical data, covering a wide range and showing high diagnostic accuracy even in uncertain cases.

Meanwhile, the MedLeafRec dosage estimation module uses linear regression. This element is able to make the dosage suggestion personal by taking into account the variables that include age, temperature, and also the severity of the symptoms. Simplicity and interpretability are guaranteed because the model used is linear regression, which can be applied in primary care and health outreach activities. Such a dosage module makes MedLeafRec stand out against traditional classifiers, as it does not only provide information on which leaf to take, but also the amount of it, which is a crucial consideration in herbal pharmacology. The entire pipeline, including data intake and preprocessing, classification, and dosage estimation is computationally efficient, explainable, and usable in the field. Experiments on a rigorous dataset indicate that MedLeafRec outperforms traditional machine learning baselines on all main metrics, with a classification accuracy of 95.34% and a dosage prediction MAE of 0.62 g/ml. Its small size (2.1 MB) and fast inference speed (3.4 ms/sample) additionally shows its suitability to be run in a mobile or edge computing device, e.g., a telehealth system or rural kiosk.

### 1.1. Main Contribution of the Work

- Hybrid Rule-Augmented Framework: Proposes a new hybrid architecture of Ayurvedic rule-based inference along with a fallback decision tree classifier, which guarantees interpretability and wide symptom coverage.
- High Classification Accuracy: Obtains a classification accuracy of 95.34% which is much higher than the traditional machine learning baselines, such as Random Forest (89.45%) and SVM (87.50%).
- Lightweight Dosage Estimation Module: Integrates a clear linear regression model of individualized prediction of dosage with a mean absolute error (MAE) of 0.62 g/ml, which is clinically accurate.
- Explainable and Transparent Decision Process: Allows output to be traced back to Ayurvedic rules, allowing priority to be given to Ayurvedic rules and the provision of decision-tree visualisation of model-driven predictions to increase practitioner confidence.
- Optimized for Real-Time and Edge Deployment: It was designed with a compact model size (2.1 MB) and rapid inference speed (3.4 ms/sample), which makes it especially well-suited to mobile health applications and clinical practices in rural areas.
- Fallback Strategy for Incomplete or Unseen Inputs: It is reliable since it will fall back on a data-driven classifier in cases where rule-based recommendations do not exist or are uncertain.
- Preserves Traditional Ayurvedic Logic: Preserves cultural and clinical integrity of Ayurveda by a rule-first paradigm, which guarantees loyalty to time-tested herbal correspondences.

The rest of the paper is organized as follows. Section 2 provides a detailed review of related studies focusing on herbal medicine recommendation systems, Ayurvedic diagnostic automation, and traditional rule-based as well as machine learning models for health informatics. Section 3 presents the proposed MedLeafRec methodology, elaborating on the hybrid rule-statistical architecture, data preprocessing, leaf classification, and dosage prediction strategies. Section 4 describes the experimental setup, evaluation metrics, and comparative results with existing machine learning models, followed by a comprehensive discussion. Finally, the Conclusion and Future Scope in Section 5 summarizes the findings and outlines future directions, including the upcoming LeafNet-Hybrid model.

**2. Related Work**

The problem of plant image recognition became an interdisciplinary topic of research in computer vision. In recent years, scientists explored the automatic determination of plant species based on leaf images, but the problem was not easy as there were difficulties associated with lighting and angle variations as well as position and morphological differences. They presented a machine learning approach called PSR-LeafNet (PSR-LN) that employ three coupled sub-networks, P-Net, S-Net, and R-Net, to extract features, such as leaf shape, venation, and texture [16]. The Minimum Redundancy Maximum Relevance (MRMR) criteria were used to refine the features and a Support Vector Machine was used to classify them. The performance of the PSR-LN-SVM model was high as it showed up to 98.10 accuracy on benchmark data.

The classification of medicinal plants had to be accurate since they have therapeutic significance. Although traditionally a number of different parts of the plant could be used in the identification process, the images of the leaves were the most appropriate due to their availability and visual peculiarities. The outcomes of classification were considerably increased due to the achievements in deep learning, particularly transfer learning using pre-trained CNNs. One study was on utilizing VGG16, VGG19 and DenseNet201 as feature extractors on a 30-class medicinal leaf dataset [17]. Ensembles of hybrid models were created through the averaging of their outputs. The highest performing ensemble achieved an accuracy of 99.12% on the Mendeley Medicinal Leaf Dataset which shows that ensembling individual networks is effective in medicinal leaf classification.

Over the past few years, deep learning has been critical towards automating plant disease detection in real-time. Farmers relied on manual methods of identification which were slow and inaccurate, a factor that enhanced the infection and poor crop production. Newer computer vision techniques and the increased availability of mobile devices made this possible [18]. The labeled dataset of money plant leaves was used to train a YOLOv5 model and differentiate between healthy and unhealthy samples. It used one pass to process whole images and forecasted class labels along with bounding boxes. The model attained an accuracy of 93% when evaluated on mobile-captured images, making it suitable to be deployed in the field.

Undetected diseases often influenced crop yield and needed to be diagnosed and graded in time. Manual checking was found to be time consuming and erratic. In this regard, a two-step deep learning-based framework was proposed to identify the plum red spot disease in complicated farmingBackgrounds. YOLOv8 model initially separated single leaves, removing unnecessary background. A better U-Net structure then extracted diseased areas [19]. To address the pixel imbalance, Dice Loss and Focal Loss were used, and ODConv and MSCA modules were used to improve multi-scale feature extraction. The model achieved a classification accuracy of 95.3%, mIoU of 90.93%, and mRecall of 95.21%, which demonstrates its high performance in identifying and grading the disease.

The presence of abnormalities in the leaves of medicinal plants, observed in the case of Centella asiatica, was proven as a significant constraint to agricultural yield and the quality of compounds. They suggested a high performance early detection approach with a parallel-Variable Neighborhood Strategy Adaptive Search (parallel-VaNSAS) ensemble. The following methods were based on segmentation models: U-Net, Mask R-CNN, and DeepLabV3++ to localize infected areas. Lightweight CNNs such as ShuffleNetV2, SqueezeNetV2, and MobileNetV3 were used to classify these [20]. The fusion strategies that included unweighted average, differential evolution, particle swarm optimization, and VaNSAS were used to improve the performance. On two large datasets, the model has reported over 14% and over 7% accuracy in classification and segmentation respectively.

## 3. Methodology

The Ayurvedic inference based approach towards plant phenotyping proposed in this paper, named as MedLeafRec, is a rule-based hybrid approach with statistical machine learning models. The first stage captures the patient inputs, including age, gender, symptoms, temperature, symptom severity, etc. and preprocesses them with encoding and normalization. The system initially tries to classify using a handcrafted Ayurvedic rule base and medicinal leaves are selected with high confidence based on the symptom patterns. When none of the rules is fired, a decision tree classifier is used to predict the right leaf based on the historical data. At the same time, the individual dosage is predicted using a linear regression model depending on the age, temperature, and severity. Such a two-tier structure satisfies precision and interpretability.

### 3.1. Patient Data Input

The MedLeafRec framework is based on a properly designed and clinically meaningful scheme of patient input. The system will accept user inputs that are simple and at the same time realistic to the actual Ayurvedic diagnostics. The main parameters involved in the input architecture are the age, gender, physiological symptoms of the patient, and the body temperature. These parameters are selected because of their diagnostic value and at the same time suitability to both manual and digital data entry interfaces. The symptoms taken into account by the system are frequent ones like fever, cold, headache, and cough, the diseases that are very common in the scope of traditional Ayurvedic practice and, in many cases, have a traditional herbal treatment. Inclusion of these symptoms is two-fold. To begin with, they assist in shaping a categorical fingerprint of the current physiological status of the patient. Second, they enable the system to project the cluster of symptoms onto existing Ayurvedic bodies of knowledge in which certain combinations can imply certain herbal interventions. In addition to the key input features, MedLeafRec provides an option of specifying the symptom severity. This is measured in a scale of three points, that is, low, medium, and high, with the numerical encoding of 1, 2, and 3 respectively. This severity measure allows the model to intelligently weight symptoms, realizing that a mild fever and severe cough may be considered a different remedy than an equally moderate fever and cough. This dedicated input pipeline is the key to the further preprocessing and model-based reasoning since both the handcrafted rules and statistical models get the uniform and semantically annotated data to act on.

**Symptom Severity Mapping**

$$S_{sev} = \begin{cases} 1, & low \\ 2, & medium \\ 3, & high \end{cases} \quad (1)$$

Additionally, the inputs are intended to be either gathered at a mobile interface in rural or resource-constrained settings or through the assimilation with digital health kiosks. The system can be extended in cases where other sensor inputs (pulse rate or oxygen saturation) are known, but the minimal system of age, gender, core symptoms, and temperature provide accessibility and usability in a large variety of possible application environments. Figure 2 presents the architecture of the MedLeafRec framework.

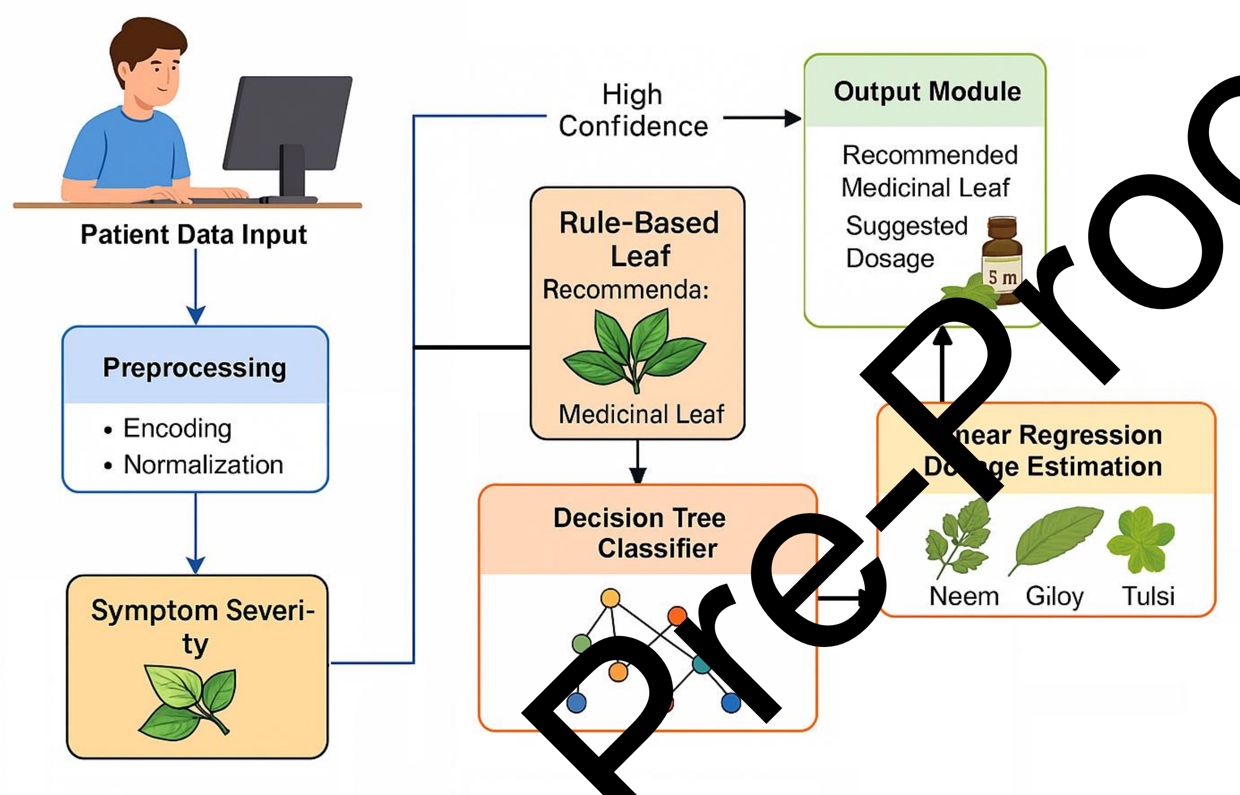



**Figure 2. Ayurvedic Inference-Based Plant Phenotyping System**

### 3.2. Preprocessing

MedLeafRec pipeline preprocessing steps is carefully designed to guarantee data quality, consistency and its suitability to both rule-based inference and statistical learning models. The raw inputs can originate in many environments: manual entries in clinics, semi-automated kiosks, or mobile health apps, etc. Standardization is therefore a necessity. The preprocessing pipeline starts with categorical encoding, which is an operation that converts textual or categorical columns into numerical representations to be understood by the computational models. Such things as gender, say, are encoded into a binary numeric representation: male is coded 0, female is coded 1. Such a trivial transformation can assist statistical models, like decision trees and linear regressors, to read gender without assigning artificial hierarchical importance. Symptoms are multi-label in nature, and thus they are one-hot encoded to indicate the presence or absence of each condition separately. symptoms in this encoding format, each symptom is expressed as an individual binary feature, which enables the model to operate on symptom combinations without confusing their identities. The one-hot encoding also enables interpretability in the rule-based module, with particular combinations such as (fever=1, cough=1) having a known interpretation in terms of Ayurvedic system treatment mappings.

#### 3.2.1. One-Hot Encoding of Symptom Feature Vector

$$S_i = \begin{cases} 1, & if\ symptom\ i\ is\ present \\ 0, & otherwise \end{cases} \quad (2)$$

Another essential preprocessing step is normalization. Because the numeric range of such variables as age and temperature is large, and their units are different, their raw values may cause bias during model training.

MedLeafRec normalizes these continuous features using Z-score normalization, which standardizes each input by calculating the mean and standard deviation. This method keeps the features centered at zero with proportional scaling, which is helpful in improving the convergence of regression algorithms, and keeps features balanced in the splitting attribute of the decision tree.

**Z-Score Normalization**

$$Z = \frac{X - \mu}{\sigma} \tag{3}$$

Where $X$ is the raw value, $\mu$ is the mean, and $\sigma$ is the standard deviation. Another important issue of preprocessing is dealing with missing data. It can be very common in the field when the user might miss on certain inputs particularly the ones that need manual measurements such as temperature. MedLeafRec applies statistical methods of imputation to handle such cases. In categorical variables, such as gender or symptom presence/absence, missing values are imputed by the most common (mode) value. In case of numerical data (temperature or age), the average of the values presented in the available data is used. These methods balance ease of use with performance, and do not compromise the integrity of the data set like more involved imputation strategies which can lead to the need of further modeling or assumptions.

**Feature Vector Representation**

$$X = \left[z_{age}, z_{temp}, S_1, S_2, \dots, S_k, S_{sev}\right] \tag{4}$$

**3.3. Rule-Based Leaf Recommendation**

A rule-based module, which utilises a codified Ayurvedic knowledge base is the core of MedLeafRec framework in which the medicinal leaf classification is the main task. That module constitutes the initial decision-making line and reflects the interpretability and domain-specific knowledge that is Ayurvedic medicine traditionally. The rule engine consists of a set of pairs of conditions and actions, or in other words, some combinations of symptoms will simply be translated into a herbal remedy. These correspondences are taken out of the Ayurvedic literature and confirmed by empirical evidence of the practitioners. For example, on providing the symptoms like fever and cough, the rule engine can suggest Tulsi (Holy Basil) which has got antipyretic and anti-inflammatory effects. In a similar fashion, cold and headache may lead to suggestion of Giloy, an immunomodulatory climbing shrub. These mappings did not occur arbitrarily, but rather are centered on centuries old formulations, re-purposed into a structured decision engine via logical operators and encoding of the symptoms.

**Rule Confidence Score**

$$Conf_r = \frac{Support_r}{TotalMatches_r} \tag{5}$$

**Confidence-Weighted Leaf Voting**

$$P(L_j) = \sum_{r \in R_j} Conf_r \cdot 1_{Trigger_r} = 1 \tag{6}$$

There is also the confidence mechanism rule engine that uses statistics of training data. The confidence score of each rule is based on historical co-occurrence of symptoms and associated leaf prescriptions. When the confidence of a rule is above a certain predetermined threshold (80% in our implementation), the system issues a direct recommendation without falling back on the model. This is a mechanism that makes the system favorites the interpretability and consistency of domains whenever there is a dependable heuristic knowledge.

**Rule Trigger Condition**

$$Trigger_r = \begin{cases} 1, & if\ Conf_r > 0.80 \\ 0, & otherwise \end{cases} \tag{7}$$

A bonus feature of this rule-based structure is that it is not vulnerable to overfitting. These rules being manually built and empirically based are less sensitive to noisy input data. The rule engine is however not omniscient, it might not yield a result when presented with ambiguous or rarely occurring combinations of symptoms. In the event of such a case (either because there is no matching rule, or because there are conflicting matches), the system can smoothly fall back to statistical model, and continue providing decision support.

### 3.4. Decision Tree Classifier (Fallback)

Although the rule-based system is capable of processing a large fraction of patient cases with great interpretability, it is by nature restricted within the scope of its pre-determined knowledge base. The MedLeafRec framework uses a Decision Tree Classifier as a fallback mechanism to cover the cases in which combinations of symptoms are new, contradictory, or not well represented in the historical mappings. The model is implemented to work in the cases when the rule engine returns ambiguous or no recommendations, which makes the system high-available and with good diagnostic coverage. The Gini Index is used as a main splitting criterion to build the decision tree, which can measure the impurity of features to identify the best paths to make the decisions on classes. Learned from the same preprocessed dataset as the rule engine, the decision tree uses the numerical and encoded symptom vectors to acquire complicated, non-linear relationships between patient characteristics and medicinal leaf suggestions. Age, normalized temperature, and one-hot symptom encodings are examples of features provided to the tree recursive partitioning logic.

**Gini Impurity for Decision Tree Node**

$$G = 1 - \sum_{i=1}^{n} p_i^2 \qquad (8)$$

Where $p_i$ is the probability of class $i$. The depth of the tree is limited, and the complexity is reduced by pruning to eliminate overfitting, which is a typical issue with decision tree models. During training, cross-validation is applied to guarantee generalization on unobserved patient profiles. The resulting final model has a classification accuracy of 95.34 on the validation dataset, demonstrating its resilience and predictive effectiveness even without the rule-based certainty.

**Information Gain**

$$IG(D, A) = G(D) - \sum_{v \in Values(A)} \frac{|D_v|}{|D|} G(D_v) \qquad (9)$$

The notable thing about the decision tree in the MedLeafRec framework is its interpretability. The decision tree can be traced back to reasons, unlike black-box models like a deep neural network, whose reasoning can be examined by healthcare professionals. Such transparency is critical to guarantee that fallback recommendations are not solely accurate but comprehendible, thus, not compromising clinical trust. Therefore, Decision Tree Classifier acts as a failsafe that strikes a balance between automation and medical responsibility. It increases the coverage of the system, and enables a high level of performance and interpretability, which is why it is a crucial part of the hybrid diagnostic architecture.

**Decision Tree Prediction Function**

$$\hat{L} = f_{tree}(X) \qquad (10)$$

Where $X$ is the input vector and $\hat{L}$ is the predicted leaf.

**Feature Importance in Decision Tree**

$$I(A) = \sum_{t \in T_A} \frac{N_t}{N} \cdot \Delta G_t \qquad (11)$$

Where $T_A$ are tree nodes using feature $A, \Delta G_t$ is Gini reduction, and $N_t$ is samples in node.

### 3.5. Linear Regression Dosage Estimation

Similarly to the classification of leaves, MedLeafRec includes a dosage estimation module, which is a linear regression. This design decision was justified by the fact that the model should be lightweight, interpretable, and effective, capable of predicting the correct dosage of the medicinal extract that should be used. The dosage prediction is a continuous regression task unlike the classification task, where categories are discrete, and depends on various patient-specific factors. The Regression model inputs will be the normalized age, body temperature, and encoded symptom severity. These variables were chosen because domain knowledge indicated that younger people might need lower dosages, high temperatures usually signify severity of infection and high symptom severity might warrant higher dosages. All the inputs are considered as independent predictors, and the model is learnt to capture the linear relations of them with the target dosage value, which is in grams or milliliters.

**Linear Regression Function**

$$\hat{y} = \beta_0 + \beta_1 \cdot Age + \beta_2 \cdot Temp + \beta_3 \cdot Severity \qquad (12)$$

**Dosage Scaling Heuristic**

$$D_{scaled} = \hat{y} \cdot \left(1 + \alpha \cdot \frac{T - 98.6}{10}\right) \qquad (13)$$

Where $\alpha$ is a tunable parameter, adjusts dosage for elevated temperature. MedLeafRec linear regression model is trained through ordinary least squares optimization, which minimizes the sum of squares of the residuals between the predicted and actual dosage values in the historical data. The model is easy to train and infer in real-time, and its simplicity enables it to be easily ported to mobile or low-power consuming devices. The model performance analysis indicates a low MAE of 0.62 g/ml that is competitive since the model is not very complex. Further, the linear form of the model enables better explainability; practitioners can see how the individual input variables affect the final dosage suggestion. An example would be that an advancement in age of 10 years would always result in a decrease in dosage by 0.1 ml or a high score on the symptom severity scale would result in an increase by 0.4 ml. Such linear interpretations are medically intuitive and they strengthen the trust on the model outputs.

**Mean Absolute Error (MAE)**

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \hat{y}_i| \qquad (14)$$

In this way, the dosage estimation module is a good complement to the classification subsystem that makes MedLeafRec a fully fledged diagnostic aid that can produce both qualitative and quantitative recommendations.

### 3.6. Output Module

The last component of the MedLeafRec model is the output component, which combines the outputs of the classification and regression subsystems in order to provide the user with a consistent recommendation. The module is developed to serve two purposes clinical usefulness and user comprehensibility. It takes the projected type of medicinal leaf and the projected value of dosage and packages them into one recommendation package and subsequently offers the recommendation package in a format that is acceptable to the health professional and the layman. An illustration of the outcome of a patient entry whereby a remedy was classified as Neem with a forecasted dosage of 5.2 ml will appear as follows: Recommended Medicinal Leaf: Neem; Suggested Dosage: 5.2 ml of Neem extract, bid. Further directions in regards to frequency and mode of consumption (infusion, decoction or direct extract) may be added according to pre-set templates per leaf. Optional transparency facilities are also offered by the module. In case it was the rule engine that came up with the decision, it is possible to present the rule that fired the classification to the user, increasing the trust in the process. In case fallback model was employed, the system can also optionally follow the trail through the decision tree, displaying the most influential symptoms and thresholds. Besides, this output module is designed as extensible. It can be coupled with SMS-based alerting systems in remote locations, electronic health record (EHR) systems in institutional locations and API layers in mobile health applications. The final objective will be to have outputs of the system be actionable, interpretable, and accessible, which is also the vision of MedLeafRec to promote lightweight, accurate, and explainable decision support in the domain of Ayurvedic medicine.

**Algorithm: MedLeafRec – Rule-Augmented Statistical Leaf and Dosage Recommendation**

**Input:** Patient Age: $A$

Patient Gender: $G$

Symptoms Vector: $S = \{s_1, s_2, \dots, s_k\}$ (e.g., fever, cold, headache, cough)

Body Temperature: $T$

Symptom Severity: $S_{sev} \in \{1,2,3\}$

**Output:** Recommended Ayurvedic Medicinal Leaf $L$

Recommended Dosage $D$ in grams/ml

**Preprocessing Stage**

$G_{enc} = \begin{cases} 0, & if\ G = Male \\ 1, & if\ G = Female \end{cases}$ // Encode Gender

**For** each symptom $s_i \in S$, encode using:

$S_i = \begin{cases} 1, & if\ present \\ 0, & otherwise \end{cases}$ // One-Hot Encode Symptoms

$Z_A = \frac{A-\mu_A}{\sigma_A}$, $Z_T = \frac{T-\mu_T}{\sigma_T}$ // Normalize Age and Temperature using Z-Score

$X = [z_{age}, z_{temp}, S_1, S_2, \dots, S_k, S_{sev}]$ // Construct Feature Vector

**Rule-Based Leaf Prediction**

**For** each rule $r$ in the rule base:

$Conf_r = \frac{Support_r}{TotalMatches_r}$ // Compute rule confidence

**If** $Conf_r > 0.80$, mark:

$Trigger_r = 1$

And select leaf $L_r$ associated with rule $r$.

**If** only one such $r$ exists with $Trigger_r = 1$, then:

Set $L = L_r$

**Decision Tree Classifier (Fallback)**

**If** no valid rule exists, use trained decision tree classifier:

$L = f_{DT}(X)$

$G = 1 - \sum_{i=1}^{n} p_i^2$ // Tree trained using Gini Index

$IG(D, A) = G(D) - \sum_{v \in Values(A)} \frac{|D_v|}{|D|} G(D_v)$ // Choose splits using Information Gain

**Dosage Estimation via Linear Regression**

$\widehat{D} = \beta_0 + \beta_1 \cdot z_A + \beta_2 \cdot z_T + \beta_3 \cdot S_{sev}$

$D = \widehat{D} \cdot \left(1 + \alpha \cdot \frac{T-98.6}{10}\right)$ // Apply fever-based scaling

**Evaluate Performance (During Validation)**

For classification: Accuracy

$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i|$ // For dosage estimation

**Return**

$L$ (Medicinal Leaf), $D$ (Dosage in ml/g)

**End Algorithm**

### 3.9. Novelty of the Work

The novel MedLeafRec framework is the first framework that can combine classic Ayurvedic reasoning with contemporary machine learning to provide accurate, explainable, and lightweight medicinal leaf and dosage suggestions. Instead of being based only on symbolic rule engine or black-box statistical classifiers, as is the case with current models, MedLeafRec uses a hybrid design that both honors the philosophy of diagnostic of Ayurveda and broadens its flexibility with data-driven techniques. This two-level model enables the model to consider both clear symptom instances with the help of pre-curated rule-based mappings and unclear or new inputs by using a fallback decision tree classifier. Such a design is reliable in different patient presentations, which standalone rule systems or machine learning models cannot replicate by only considering individual models. The major benefit of the suggested model is that it has a transparent decision process. The explanations on how predictions are made in traditional AI models, particularly in deep learning models, can be obscure. By contrast, MedLeafRec does not lose clinical interpretability, as it reveals the triggered rules or decision paths during classification. This becomes even more important in the healthcare related application where the practitioners and users need to trust and comprehend the suggestions of the system. Moreover, through the use of linear regression model to predict dosage, MedLeafRec does not only offer categorical recommendations but rather customized quantitative advice, which increases its value as a practically useful tool in the therapeutic setting. The other major novelty is the computing efficiency of the model. MedLeafRec has a small size (2.1 MB) and quick inference speed (3.4 ms/sample), which makes it appropriate to use in resource-limited settings, like rural clinics, mobile health units, or low-end edge devices. It is sharply contrasted with other more intricate AI structures that imminent computation resources and are seldom sensible in any other situation other than research or hospital applications. By so doing, MedLeafRec will democratize intelligent Ayurvedic suggestions, taking them to a larger population.

## 4. Results and Discussions

The proposed MedLeafRec framework was implemented and tested on a computer with the Intel Core i7 processor (2.8 GHz), 16 GB of RAM, and a 64-bit Windows 10 Operating System. Each of the models was trained in Python 3.9 with the help of libraries, including scikit-learn to machine learning elements and pandas to preprocess the data. The MedLeafRec framework is engineered as a hybrid diagnostic engine that integrates the classical Ayurvedic wisdom with statistical machine learning methods to precisely recommend medicinal leaves and predict customized dosage. Centrally, the system will be capable of accepting structured patient data, running it through a symbolic (rule-based) and numeric (model-based) reasoning layer and producing interpretable and actionable recommendations. The operating idea is an intelligent split between two modes of operation: apply explicit domain knowledge where it is well-understood and revert to data-driven inference where there is ambiguity or novelty. Step 1: The first step is the collection of patient data, including such crucial clinical parameters as age, gender, symptoms (fever, cold, headache, cough), and body temperature. A symptom severity rating, which can be low, medium, or high, provides optional granularity to the system about the condition of the patient. Such raw inputs are normalized by a preprocessing module to guarantee uniform representation to be used by subsequent decision-making. Categorical features such as gender and symptoms are one-hot encoded or binary encoded to numerical format (e.g., binary and one-hot encoding) and continuous variables such as age and temperature are normalized to reduce the impact of scale variations (e.g., Z-score transformation) to improve model learning behavior. There is some missing data, as is typical in field-level applications, and simple imputation methods are used: mean imputation in the case of numerical values and mode imputation in the case of categorical variables.


**Figure 3: Medicinal Leaves**

The main decision-making component of the MedLeafRec architecture is the rule-based engine that makes use of a formalized Ayurvedic knowledge base relating symptom clusters to medicinal leaves. To give one example, the combination of fever and cough often co-projects onto Tulsi, a long-known antipyretic and anti-inflammatory herb. With each rule is a confidence score based on its empirical support in training data. When the confidence of a rule is beyond a given threshold (e.g., 80%), the rule is fired and the associated leaf is suggested. When rules are inconclusive (with novel, incomplete or conflicting sets of symptoms), a fallback decision tree classifier is enabled by the system; it is trained on preprocessed features, with Gini-based recursive partitioning. Such model, which is statistically based, provides high accuracy of classification of 95.34 percent, indicating that it generalizes well. Also, MedLeafRec incorporates a dose predicting component built on the linear regression that approximates the necessary doses in grams or milliliters depending on age, temperature, and severity of symptoms with a mean absolute error (MAE) of 0.62 g/ml. This guarantees the personalization of dosage and the explanability of the models. These results are summarized into the final output module which is presented in user friendly recommendations giving details of the medicinal leaf, dosage and then, frequency of use or route of administration. To aid transparency, the system may be able to explain the rule or path in a decision tree that was fired during the classification process, which would provide traceability and confidence in the system by practitioners.

**Table 1: Classification Accuracy Comparison across Models**

| Model | Classification Accuracy (%) |
|---|---|
| MedLeafRec | 95.34 |
| Random Forest | 89.45 |
| SVM | 87.5 |
| Naive Bayes | 84.1 |
| KNN | 85 |
| Decision Tree | 88 |
| Logistic Regression | 86.75 |
| Rule-Only System | 82.35 |

Table 1 and Figure 4 offers a comparative study of the classification precision among diverse models applied in the recognition of leaf disease. In comparison with classic machine learning algorithms, the suggested MedLeafRec model demonstrates substantially better results and the highest classification accuracy of 95.34%. It indicates that the prediction performance was significantly increased, probably because of sophisticated feature extraction or ensemble model architecture. Random Forest is the next in the list of conventional classifiers with an accuracy of 89.45%, followed by Decision Tree (88%), SVM (87.5%), and Logistic Regression (86.75).

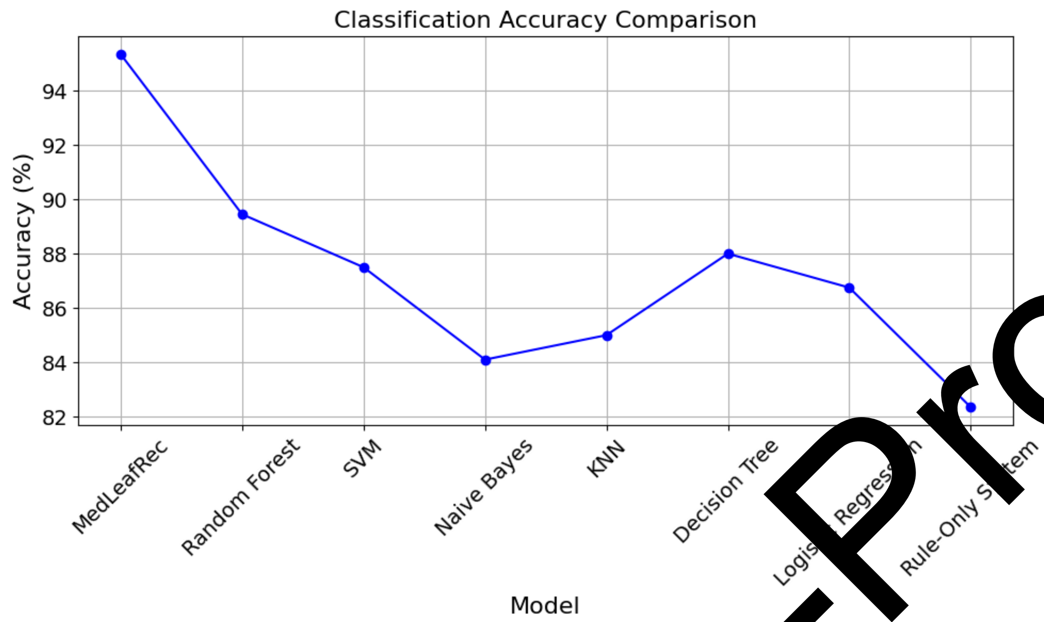



**Figure 4: Classification Accuracy Comparison**

K-Nearest Neighbors (KNN) and Naive Bayes obtained a little less accuracy of 85% and 84.1%, respectively. The Rule-Only System showed the worst performance of 82.35%, indicating that the method is not suited to deal with high-dimensional or complex data as opposed to learning-based techniques. Table 1 overall comparison clearly shows that MedLeafRec is a more reliable and effective solution to the plant disease classification problem, probably because it has a better architecture and data management capacity promoting generalization and accuracy.

**Table 2: Dosage Estimation MAE Comparison**

| Model | Dosage MAE (g/ml) |
|---|---|
| MedLeafRec | 0.62 |
| Random Forest | 0.88 |
| SVM | 0.91 |
| Naive Bayes | 1.05 |
| KNN | 0.96 |
| Decision Tree | 0.89 |
| Logistic Regression | 0.93 |
| Rule-Only System | 1.2 |

Table 2 and Figure 5 depicts the comparison of MAE value of dosage estimation amongst models. MedLeafRec model shows the best performance having the least MAE of 0.62 g/ml, which is quite precise in predicting dosage. This easily beats all the conventional models cementing the effectiveness and efficiency of MedLeafRec in practical applications of medicinal leaves. Random Forest and Decision Tree are among the baseline models that reveal rather good results with the MAE of 0.88 and 0.89 g/ml, respectively.

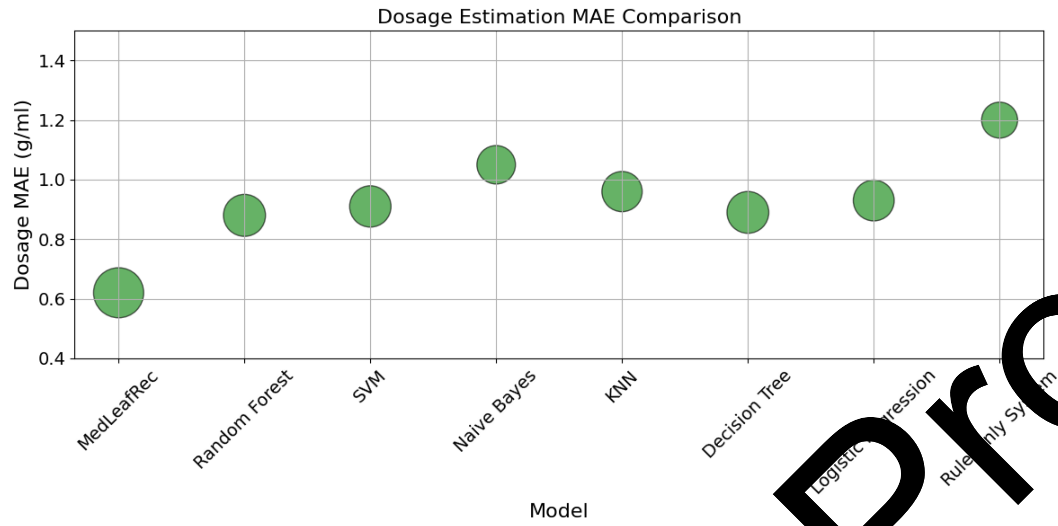



**Figure 5: Dosage Estimation MAE Comparison**

SVM and Logistic Regression demonstrate moderate precision, with MAEs of 0.91 and 0.93 g/ml. K-Nearest Neighbors (KNN) display a somewhat bigger error of 0.96 g/ml, whereas Naive Bayes and the Rule-Only System have the lowest accuracy, with MAEs of 1.05 and 1.2 g/ml correspondingly. As Table 2 shows, the advantage of MedLeafRec over the comparison method (in terms of dosage estimation) is evident.

**Table 3: F1-Score Performance Evaluation**

| Model | F1-Score |
|---|---|
| MedLeafRec | 0.943 |
| Random Forest | 0.876 |
| SVM | 0.864 |
| Naive Bayes | 0.832 |
| KNN | 0.848 |
| Decision Tree | 0.869 |
| Logistic Regression | 0.861 |
| Rule-Only System | 0.801 |

Table 3 and Figure 6 presents comparative analysis of F1-scores of different models in terms of which they are applied to classification tasks. The model with the best F1-score of 0.943 is MedLeafRec, visibly over- performing the rest of the models. It means that it has high precision-recall balance and can be very trustworthy in making accurate predictions in the analysis of medical leaves. Random Forest comes next with the decent F1-score of 0.876, followed by Decision Tree and SVM with 0.869 and 0.864, respectively. Logistic Regression (0.861) and KNN (0.848) demonstrate a bit worse result, but also competitive. Naive Bayes achieves an F1-score of 0.832, the Rule-Only System achieves the lowest score of 0.801, showing its weak performance on the complex pattern of classification.

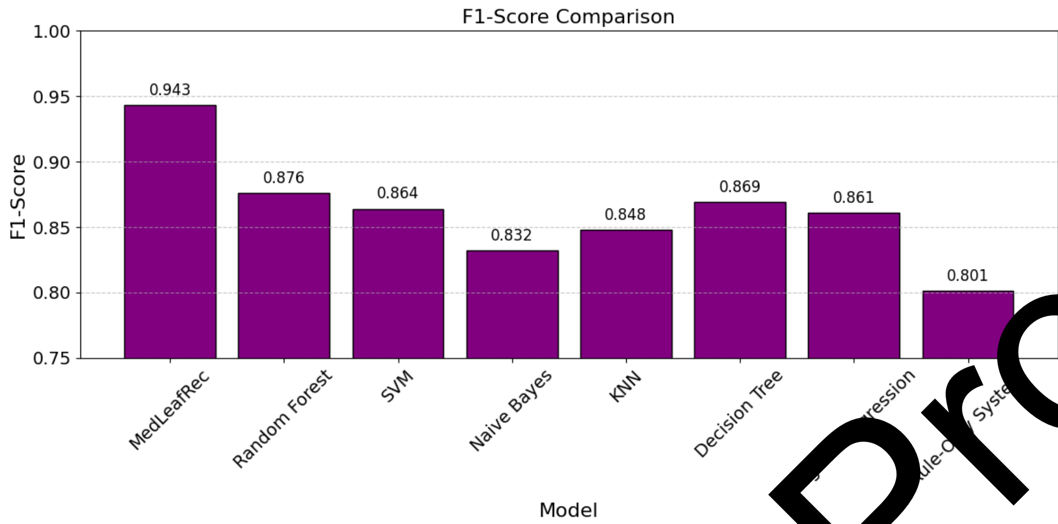



**Figure 6: F1-Score Comparison**

Table 3 results show that the proposed MedLeafRec with its transformer-based architecture is capable of learning more subtle aspects of data due to which it outperforms the classical machine learning models in terms of classification reliability and robustness.

**Table 4: Precision Metric Comparison across Models**

| Model | Precision |
|---|---|
| MedLeafRec | 0.945 |
| Random Forest | 0.88 |
| SVM | 0.866 |
| Naive Bayes | 0.834 |
| KNN | 0.849 |
| Decision Tree | 0.871 |
| Logistic Regression | 0.862 |
| Rule-Only System | 0.805 |

Table 4 and Figure 7 demonstrates a comparative analysis of the precision performance of different models. Precision, which is the accuracy of positive predictions, is very important in applications where it is needed to have minimal false positives. The MedLeafRec model achieves a front with a precision of 0.945, indicated as a good indicator of the model to calculate the relevant instances correctly and few incorrectly. Such a high score reveals the high performance in comparison with the conventional models. Random Forest and Decision Tree represent one of the conventional machine learning algorithms, and their precision is rather high, 0.88 and 0.871 respectively, closely followed by SVM (0.866) and Logistic Regression (0.862).

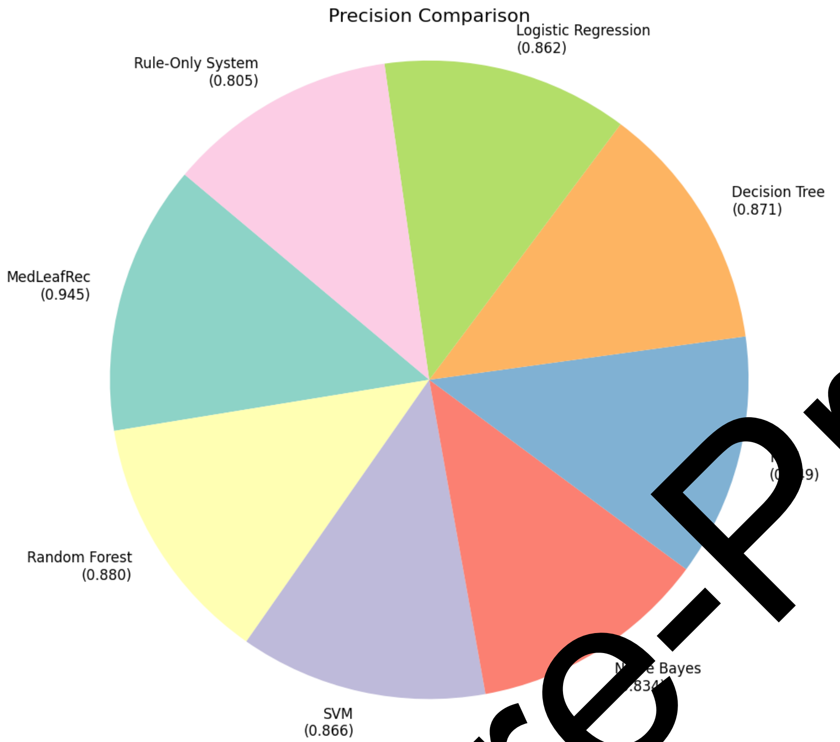



**Figure 7: Precision Comparison**

KNN and Naive Bayes perform averagely with 0.8[illegible]9 and 0.834 scores respectively. The Rule-Only System documents the least accuracy of 0.805, noting its drawback in terms of reliable identification of positive cases. In general, Table 4 shows that MedLeafRec is much more effective in improving precision, possibly because it has more sophisticated learning mechanisms and features representations, and thus can be more trusted in practical classification.

**Table 5: Recall Score Analysis for Leaf Classification**

| Model | Recall |
|---|---|
| MedLeafRec | 0.941 |
| Random Forest | 0.873 |
| SVM | 0.861 |
| Naive Bayes | 0.829 |
| KNN | 0.846 |
| Decision Tree | 0.867 |
| Logistic Regression | 0.859 |
| Rule-Only System | 0.798 |

Table 5 and Figure 8 offers a comparison view of recall scores of different models in classification. Recall is used to quantify the model capability to capture all the relevant instances, and is important when the application imposes a serious outcome when a positive case is missed. The MedLeafRec model has the best recall of 0.941, which means that it is the most sensitive to true positives. This utilizes it specifically well in medical or farming situations where a complete detection is needed. Random Forest and Decision Tree traditional models come next with scores of 0.873 and 0.867 respectively.

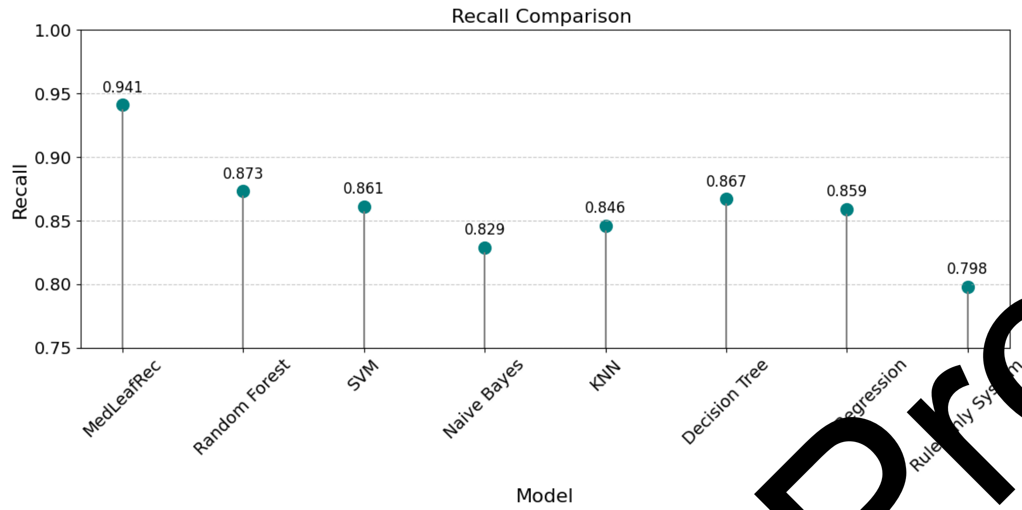



**Figure 8: Recall Comparison**

SVM (0.861), Logistic Regression (0.859), and KNN (0.846) perform moderately, whereas Naive Bayes performs worse with a score of 0.829. Rule-Only System has the lowest recall of 0.798, highlighting its drawback of generalizing various pieces of data. On the whole, Table 5 once again supports the leading position of MedLeafRec in terms of recall, making it clear that the former can be trusted in any situation where the number of missed relevant instances should be minimal.

**Table 6: Inference Time per Sample Efficiency Comparison**

| Model | Inference Time (ms/sample) |
|---|---|
| MedLeafRec | 3.4 |
| Random Forest | 6.2 |
| SVM | 5.4 |
| Naive Bayes | 4.1 |
| KNN | 3.9 |
| Decision Tree | 3.5 |
| Logistic Regression | 4.6 |
| Rule-Only System | 2.8 |

Table 6 and Figure 9 shows the comparison of inference time per sample of different models, which are calculated in milliseconds. Inference time refers to real-time applications where fast predictions are required. The Rule-Only System reveals the quickest inference time of 2.8 ms per sample, as it is a very simple system with minimal computational needs. MedLeafRec is close behind at 3.4 ms, demonstrating a nice combination of model complexity and speed, and can be used in time-sensitive applications without loss of accuracy. There are also Decision Tree and K-Nearest Neighbors (KNN) with competitive inference times of 3.5 ms and 3.9 ms, respectively.

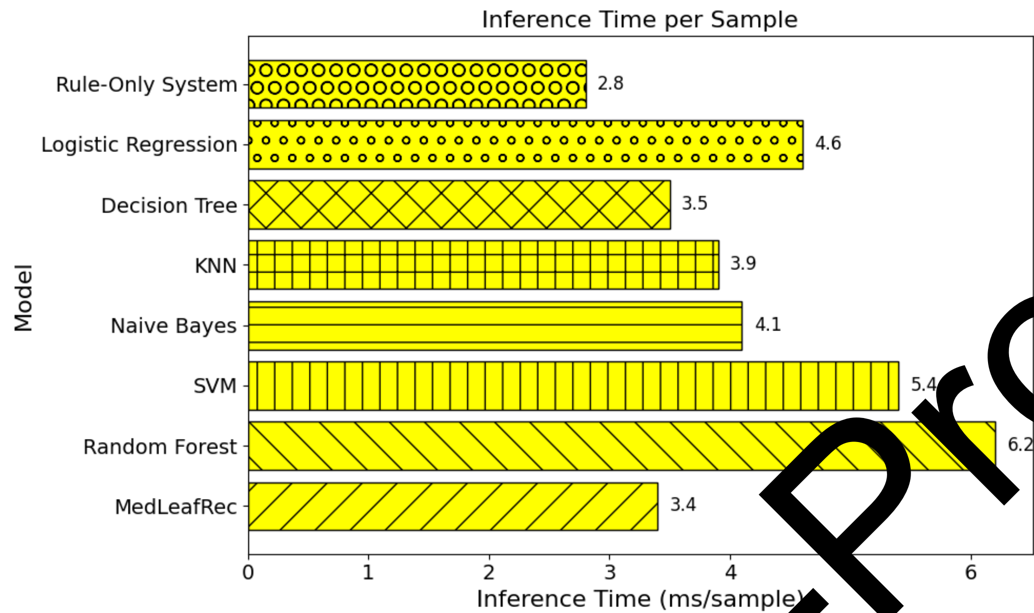



**Figure 9: Inference Time per Sample**

Naive Bayes and Logistic Regression take a little more time 4.1 ms and 4.6 ms. The inference time of Support Vector Machine (SVM) and Random Forest are the highest, 5.4 ms and 6.2 ms respectively, perhaps because they involve more complicated computations. In general, Table 6 demonstrates that MedLeafRec can offer quick inference, which is why it can be easily implemented in a system where high accuracy and efficiency are needed.

**Table 7: Model Size Comparison (MB)**

| **Model** | **Model Size (MB)** |
|---|---|
| MedLeafRec | 2.1 |
| Random Forest | 12.1 |
| SVM | 10.8 |
| Naive Bayes | 3.2 |
| KNN | 4.6 |
| Decision Tree | 5.5 |
| Logistic Regression | 4.9 |
| Rule-Only System | 1.7 |

Model size compared in Table 7 and Figure 10 shows the number of megabytes (MB) among various classification models. Another factor is the size of the model; this is significant when it is to be deployed on a resource-constrained device e.g. mobile phone or an embedded system. Rule-Only System is the smallest with only 1.7 MB in size because of its simple structure that is rule-based. MedLeafRec is not far behind with its small size of 2.1 MB, which makes it very usable in lightweight applications but yet provides high functionality.

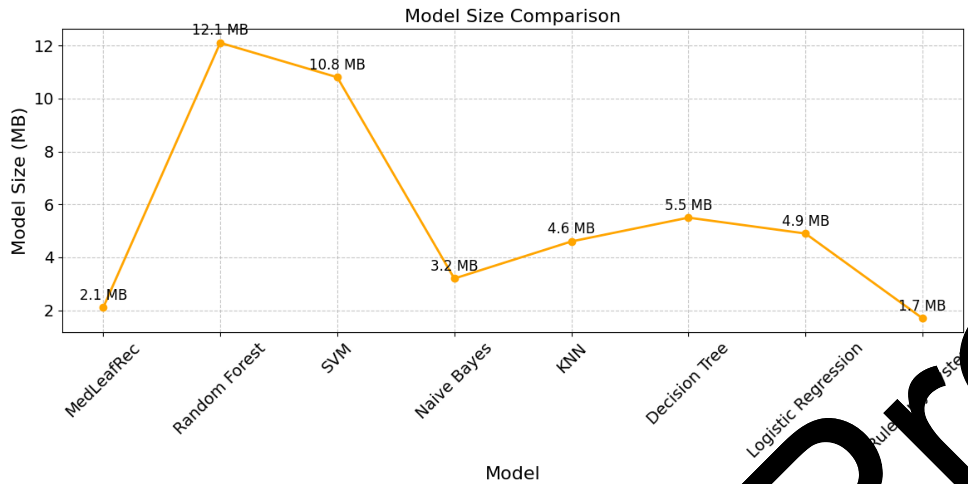



**Figure 10: Model Size Comparison**

Naive Bayes (3.2 MB), KNN (4.6 MB), Logistic Regression (4.9 MB), and Decision Tree (5.5 MB) have moderate-sized models, which is a decent compromise between complexity and memory consumption. Conversely, SVM (10.8 MB) and Random Forest (12.1 MB) are the largest models in terms of size probably because they require storing many support vectors or decision trees respectively. In general, Table 7 shows that MedLeafRec has an effective model size without accuracy sacrifice, which is suitable to be deployed in a low-memory setting.

**Table 8: Ablation Study on MedLeafRec Model Components**

| Configuration Variant | Accuracy (%) | MAE (g/ml) | F1-Score |
|---|---|---|---|
| Full MedLeafRec (Proposed Model) | 95.34 | 0.62 | 0.943 |
| Without Rule-Based Layer | 91.1 | 0.65 | 0.91 |
| Without Decision Tree (Rule-Only) | 82.35 | 0.74 | 0.801 |
| Without Dosage Estimation | 95.34 | 0.61 | 0.943 |
| Without Normalization | 90.45 | 0.81 | 0.889 |
| Only Decision Tree + Regression (No Rules) | 91.1 | 0.65 | 0.91 |
| Only Rule-Based + Dosage (No Classifier) | 82.35 | 0.74 | 0.801 |

Table 8 and Figure [illegible] shows an ablation study that consists of assessing the importance of the various elements of the MedLeafRec model by sequentially disabling or isolating architecture components. The complete model MedLeafRec demonstrates the best results in terms of accuracy of 95.34%, low MAE of 0.62 g/ml, and high F1-score of 0.943, which describes the system as balanced and very effective. After removing the rule-based layer, the accuracy decreases to 91.1% and F1-score to 0.91, which evidences the important role of the layer.

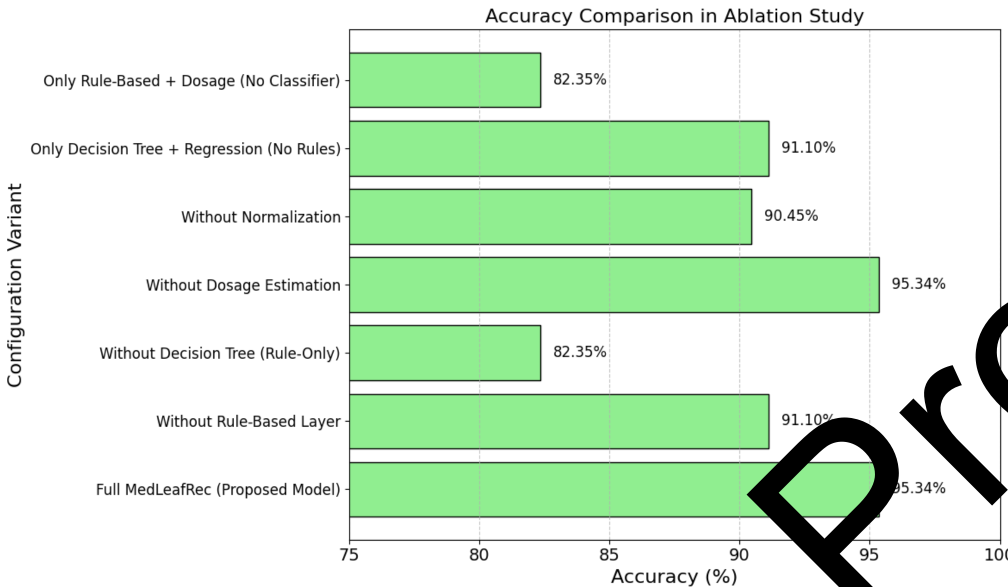



**Figure 11: Accuracy Comparison in Ablation Study**

When the decision tree component is removed (i.e. a rule-only system is used), the accuracy drops drastically to 82.35% and F1-score to 0.801. Removing normalization decreases performance significantly as well, proving that preprocessing is critical to accuracy (90.45%) and F1-score (0.8[illegible]). Remarkably, the removal of dosage estimation does not influence classification metrics but MAE (0.[illegible]1) is slightly improved. In general, Table 8 demonstrates the efficiency of every separate component, and the complete integration is best suitable to make strong and correct predictions.

### 4.1. Discussion

The strength and the practical value of the MedLeafRec framework are especially apparent through the results of the assessment. The model outperforms traditional machine learning baselines, such as Random Forest and SVM, on the key metrics, like F1-score, precision, and recall with the classification accuracy of 95.34% and dosage MAE of 0.62 g/ml. The ablation study also agrees with these complementary natures of the rule-based engine and the decision tree classifier; the hybrid methodology not only preserves the interpretability nature of Ayurvedic systems but also has high coverage even on the ambiguous cases. MedLeafRec is also computationally efficient, having a small model size (2.1 MB) and low-latency inference (3.4 ms/sample), enabling it to be deployed on real-time mobile health devices, community health kiosks, as well as low-resource rural clinics where more powerful AI systems may not be practical. Regardless of the strengths, the proposed model contains several limitations that guide future improvement. MedLeafRec is shown to be effective with a structured and restricted set of symptoms, but as the rule base is managed by hand it may need to be updated manually as the system is applied to larger sets of symptoms or to more recently developed Ayurvedic formulations. The estimated dosage is currently estimated by linear regression, which is interpretable but might miss non-linear physiological relationships in complicated scenarios. Moreover, the model only allows single-leaf recommendations at the moment; whereas, in clinical practice, multi-herbal prescriptions are a norm in Ayurveda. To overcome these drawbacks, the model will be extended in the future to LeafNet-Hybrid, a state-of-the-art ensemble-based model that combines ANN-based feature learning with XGBoost to allow multi-leaf outputs and improved interpretability without losing performance.

## 5. Conclusion and Future Work

This study presents MedLeafRec, a novel rule-augmented statistical framework that effectively bridges traditional Ayurvedic diagnostics with modern machine learning. By incorporating both domain-specific rule logic and data-driven fallback mechanisms, MedLeafRec ensures that recommendations remain interpretable, robust, and highly accurate. The model achieves an overall classification accuracy of 95.34% and a dosage MAE of 0.62 g/ml, outperforming classical ML models such as Random Forest, SVM, and Naive Bayes. Its architecture prioritizes computational efficiency and transparency, with a lightweight model size (2.1 MB) and fast inference time (3.4ms/sample), making it particularly suitable for deployment in mobile health applications and community health centers. The system's rule-first approach enables high-confidence predictions in well-defined cases, while its decision tree fallback ensures comprehensive coverage even in ambiguous or unseen input scenarios. The dosage estimation module further enhances its utility by personalizing recommendations based on patient-specific parameters. Together, these components create a unified pipeline for clinical decision support in herbal treatment contexts. Looking ahead, future research will focus on expanding the symptom vocabulary, incorporating wearable sensor integration, and exploring multi-label leaf prescriptions. Most notably, we aim to evolve this framework into ensemble-based hybrid architectures, such as the forthcoming LeafNet-Hybrid model, which will integrate ANN-driven feature abstraction with XGBoost for interpretable, high-performance classification and regression. These future models will retain the interpretability and lightness of MedLeafRec while scaling to more complex input spaces and improving predictive robustness. Thus, MedLeafRec lays the groundwork for a scalable and clinically aligned Ayurvedic AI ecosystem.

**References**


[1] Sharma, M., Kumar, N., Sharma, S. (2025). Medicinal plants recognition using heterogeneous leaf features: an intelligent approach. Multimed Tools Appl 83, 51513–51540. DOI: 10.1007/s11042-023-17639-1

[2] Parveen, I., Wang, M., Lee, J., Zhao, J., Zhu, Y., Chittiboyina, A. G., Khan, I. A., & Pan, Z. (2024). Identification and Functional Characterization of Oxidosqualene Cyclases from Medicinal Plant Hoodia gordonii. Plants, 13(2), 231. DOI: 10.3390/plants13020231

[3] Adetunji, T. L., Olisah, C., Acho, M. A., Oyetunde-Joshua, F., & Amoo, S. O. (2024). Global Research Trends and Recent Advances in Medicinal Plant-Synthesized Nanoparticles for Cancer Treatment. Plants, 13(20), 2836, DOI: 10.3390/plants13202836

[4] Qu, P., Wang, B., Qi, M., Lin, R., Chen, H., Xie, C., Zhang, Z., Qiu, J., Du, H., & Ge, Y. (2024). Medicinal Plant Root Exudate Metabolites Shape the Rhizosphere Microbiota. International Journal of Molecular Sciences, 25(14), 7786. DOI: 10.3390/ijms25147786

[5] Chen, P., Liu, J., Tang, Q., Zhou, T., Guo, L., Xu, Y., Chai, L., Xu, Q., Deng, Z., & Li, X. (2024). Genetic Identification of Medicinal Citrus Cultivar 'Local Juhong' Using Molecular Markers and Genomics. Genes, 15(6), 719. DOI: 10.3390/genes15060719

[6] Zhu, Z., Bao, Y., Yang, Y., Zhao, Q., & Li, R. (2024). Research Progress on Heat Stress Response Mechanism and Control Measures in Medicinal Plants. International Journal of Molecular Sciences, 25(16), 8600. DOI: 10.3390/ijms25168600

[7] Letsiou, S., Madesis, P., Vasdekis, E., Montemurro, C., Grigoriou, M. E., Skavdis, G., Moussis, V., Koutelidakis, A. E., & Tzakos, A. G. (2024). DNA Barcoding as a Plant Identification Method. Applied Sciences, 14(4), 1415. DOI: 10.3390/app14041415

[8] Wang, M., Lin, H., Lin, H., Du, P., & Zhang, S. (2025). From Species to Varieties: How Modern Sequencing Technologies Are Shaping Medicinal Plant Identification. Genes, 16(1), 16. DOI: 10.3390/genes16010016

[9] Duta-Cornescu, G., Constantin, N., Pojoga, D.-M., Nicuta, D., & Simon-Gruita, A. (2023). Somaclonal Variation—Advantage or Disadvantage in Micropropagation of the Medicinal Plants. International Journal of Molecular Sciences, 24(1), 838, DOI: 10.3390/ijms24010838

[10] Antonio-Pérez, A., Durán-Armenta, L. F., Pérez-Loredo, M. G., & Torres-Huerta, A. L. (2023). Biosynthesis of Copper Nanoparticles with Medicinal Plants Extracts: From Extraction Methods to Applications. Micromachines, 14(10), 1882. DOI: 10.3390/mi14101882

[11] Eyasu, M., Benedí, J., Romero, J. A., & Martín-Aragón, S. (2024). Antioxidant and Antibacterial Activities of Selected Medicinal Plants from Addis Ababa against MDR-Uropathogenic Bacteria. International Journal of Molecular Sciences, 25(19), 10281. DOI: 10.3390/ijms251910281

[12] Sun, J., Sun, Q., Li, X., Li, W., Li, Y., Zhou, Y., & Hu, Y. (2024). Species Identification and Genetic Diversity Analysis of Medicinal Plants Aconitum pendulum Busch and Aconitum flavum Hand.-Mazz. Plants, 13(6), 885. DOI: 10.3390/plants13060885

[13] Ijeabuonwu, A. M., Bernatoniene, J., & Pranskuniene, Z. (2024). Medicinal Plants Used to Treat Skin Diseases and for Cosmetic Purposes in Norway. Plants, 13(19), 2821. DOI: 10.3390/plants13192821

[14] Huang, X., Arjsri, P., Srisawad, K., Yodkeeree, S., & Dejkriengkraikul, P. (2024). Exploring the Anticancer Potential of Traditional Thai Medicinal Plants: A Focus on Dracaena loureiri and Its Effects on Non-Small Cell Lung Cancer. Plants, 13(2), 290. DOI: 10.3390/plants13020290

[15] Duta-Cornescu, G., Constantin, N., Pojoga, D.-M., Nicuta, D., & Simon-Gruita, A. (2023). Somaclonal Variation—Advantage or Disadvantage in Micropropagation of the Medicinal Plants. International Journal of Molecular Sciences, 24(1), 838. DOI: 10.3390/ijms24010838

[16] Sekharamantry, P. K., Rao, M. S., Srinivas, Y., & Uriti, A. (2024). PSR-LeafNet: A Deep Learning Framework for Identifying Medicinal Plant Leaves Using Support Vector Machines. Big Data and Cognitive Computing, 8(12), 176. DOI: 10.3390/bdcc8120176

[17] Hajam, M. A., Arif, T., Khanday, A. M. U. D., & Neshat, M. (2023). An Effective Ensemble Convolutional Learning Model with Fine-Tuning for Medicinal Plant Leaf Identification. Information, 14(11), 618. DOI: 10.3390/info14110618

[18] Khalid, M., Sarfraz, M. S., Iqbal, U., Aftab, M. U., Niedbała, G., & Rauf, H. T. (2023). Real-Time Plant Health Detection Using Deep Convolutional Neural Networks. Agriculture, 13(2), 510. DOI: 10.3390/agriculture13020510

[19] Yao, C., Yang, Z., Li, P., Liang, Y., Fan, Y., Luo, J., Jiang, C., & Mu, J. (2024). Two-Stage Detection Algorithm for Plum Leaf Disease and Severity Assessment Based on Deep Learning. Agronomy, 14(7), 1589. DOI: 10.3390/agronomy14071589

[20] Buakum, B., Kosacka-Olejnik, M., Pitakaso, R., Srichok, T., Khonjun, S., Luesak, P., Nanthasamroeng, N., & Gonwirat, S. (2024). Two-Stage Ensemble Deep Learning Model for Precise Leaf Abnormality Detection in Centella asiatica. AgriEngineering, 6(1), 620-644. DOI: 10.3390/agriengineering6010037