Development of AI – Driven Recommendation System for Herbal Medicines Based on Symptoms

Minor Project Report

Submitted for the partial fulfillment of the degree of

Bachelor of Technology

In

Information Technology

Submitted By

Unnati Jadon (0901IT231071)

Saksham Pahariya (0901IT231059)

UNDER THE SUPERVISION AND GUIDANCE OF

Dr. Punit Kumar Johari

Associate Professor

Department of Information Technology

MADHAV INSTITUTE OF TECHNOLOGY & SCIENCE, GWALIOR (M.P.), INDIA
माधव प्रौद्योगिकी एवं विज्ञान संस्थान, ग्वालियर (म.प्र.), भारत
A GOVT. AIDED UGC AUTONOMOUS INSTITUTE, AFFILIATED TO R.G.P.V. BHOPAL (M.P.), INDIA
NAAC ACCREDITED WITH A++ GRADE

July – December 2025

DECLARATION BY THE CANDIDATE

I hereby declare that the work entitled **“**Development of AI – Driven Recommendation System for Herbal Medicines Based on Symptoms” is my work, conducted under the supervision of **Dr. Punit Kumar Johari, Associate Professor,** during the session July-December 2025. The report submitted by me is a record of bonafide work carried out by me.

I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Unnati Jadon (0901IT231071)

Saksham Pahariya (0901IT231059)

Date:
Place: Gwalior

This is to certify that the above statement made by the candidates is correct to the best of my knowledge and belief.

Guided By:

Dr. Punit Kumar Johari
Associate Professor
Information Technology
MITS, Gwalior

Departmental Project Coordinator

Prof. Vikas Sejwar
Assistant Professor
Information Technology
MITS, Gwalior

Approved by HoD

Dr. Akhilesh Tiwari
HoD
Information Technology
MITS, Gwalior

PLAGIARISM CHECK CERTIFICATE

This is to certify that we, a student of B.Tech. in **Information Technology** have checked my complete report entitled **“**Development of AI – Driven Recommendation System for Herbal Medicines Based on Symptoms” for similarity/plagiarism using the “Turnitin” software available in the institute.

This is to certify that the similarity in my report is found to be …… which is within the specified limit (30%).

The full plagiarism report along with the summary is enclosed.

Unnati Jadon (0901IT231071)

Saksham Pahariya (0901IT231059)

Checked By:	**Approved By:**
--------------------------------	**--------------------------------**
Dr. Punit Kumar Johari **Associate Professor** Information Technology MITS, Gwalior	**Rajeev Kumar Singh** **Assistant Professor** Information Technology MITS, Gwalior

Abstract

AyurPredict-AI is a machine-based web-based application that interface traditional Ayurveda knowledge with contemporary computational intelligence in making evidence-based recommendations in herbs and phytochemical insights. It employs the two-way structure to laypersons who want to know how to relieve symptoms, and researchers who study the nature of herbs.

In this application, a high-performance Random Forest Regressor model is used, that is, it is optimized by using extreme feature engineering and hyperparameter optimization. Our pipeline is biologically plausible instead of the properties that other traditional models use that are prone to leakage; we design domain-specific features, including Herb-Target Interaction scores, Compound-Target Affinity measures, and Neuro-Inflammatory Balance Indicators. Model selection is not limited to a standard R 2 analysis but incorporates ad-hoc measures of trustworthiness, both in terms of prediction stability, error stability and classification performance (measured by Precision, Recall and F1-Score).

The User Pathway, which is based on a Flask backend and is responsive, accepts symptom inputs and provides ranked herb predictions, but the Researcher Pathway provides detailed information on a single herb, including a description of biological targets, confidence score, therapeutic benefits, and safety concerns. The system encodes queries using encoders with advanced features and verifies the predictions using a curated dataset to make sure that the predictions are scientific.

This publication represents the actual combination of classic medical science and the use of machine learning techniques and enables the developed tool to be accessible to both healthcare consumers and Ayurvedic researchers.

ACKNOWLEDGEMENT

I would like to express my greatest appreciation to all the individuals who have helped and supported me throughout this minor project. I am thankful to the whole Information Technology department for their ongoing support during the project, from initial advice and provision of contact in the first stages through ongoing advice and encouragement, which led to the final report of this minor project.

A special acknowledgement goes to my colleagues who help me in completing the file and by exchanging interesting ideas to deal with problems and sharing the experience.

I wish to thank **Dr. Punit Kumar Johari** as well for his undivided support and interests which inspired me and encouraged me to go my own way without whom I would be unable to complete my project.

At the end, I want to thank my friends who displayed appreciation to my work and motivated me to continue my work

----

Unnati Jadon (0901IT231071)

Saksham Pahariya (0901IT231059)

Content

Table of Contents

Acronyms and Nomenclature

AI	Artificial Intelligence
API	Application Programming Interface
CSS	Cascading Style Sheets
CSV	Comma-Separated Values
DT	Decision Tree
F1-Score	Harmonic mean of precision & recall
Flask	Python web framework
HTML	HyperText Markup Language
JSON	JavaScript Object Notation
MAE	Mean Absolute Error
ML	Machine Learning
Precision	TP / Predicted Positives
Recall	TP / Actual Positives
RF	Random Forest
RMSE	Root Mean Square Error
SMILES	Molecular structure string format
TP	True Positive
FP	False Positive
FN	False Negative
TN	True Negative

List of Figures

LIST OF TABLES

Chapter 1: Introduction

The Ayurveda has been one of the oldest surviving systems of a healing system that has been leading wellness practices that date back to more than 5000 years old. The problem of accessing this ancient wisdom is one of the greatest problems in the digital times. Individuals with regular day-to-day things are hardly knowledgeable of which ayurvedic herbs would assist them, yet scientists waste untold hours scrolling through the old literature and the contemporary findings to comprehend the characteristics of herbs and their interactions. This is where AyurPredict-AI comes in to connect ancient knowledge of ayurvedic medicine with the current machine learning possibilities.

The Challenge

Nowadays, when one is having a headache or is having some stomach troubles, he/she goes to Google, sifting through conflicting news and advertisements. In the same case, researchers have fragmented sources of data and this has contributed to the fact that it is challenging to comprehend the biological target of an herb, benefits and safety concerns fast.

Our Solution

1. AyurPredict-AI is a web application that is specifically aimed to make the information on ayurvedic herbs available, reliable, and scientifically supported.
2. This system has 2 separate user groups with well-thought pathways:
3. To common users: all they need to do is type in a symptom they are experiencing and AyurPredict-AI studies it and prescribes an appropriate ayurvedic herb that can help. There is no necessity to learn Sanskrit names and read medical journals.
4. To the Researchers and Practitioners: Type any ayurvedic herb name, the system would offer detailed information of the biological targets, confidence scores, therapeutic effects and details of significant safety information. It is comparable to having a research assistant.

What Makes It Special?

Compared to basic database searches or keywords matching database systems, AyurPredict-AI involves a well-developed ML model. We did not just feed an algorithm with data and hope it would work. Rather, we designed aspects that reflect real herb-target interactions, compound affinities and therapeutic balance indicators so that predictions would occur in real scientific relations. The model was not only tested in terms of accuracy but also trustworthiness. We did stability, consistency of errors, and biological soundness.

Technical Foundation

Programmed with Flask backend and a sleek and responsive front end, the application handles the queries by executing advanced features encoders and Random Forest Regressor model trained with the help of hyperparameter optimization,

Chapter 2: Literature Survey

Machine Learning coupled with traditional ayurvedic medicine in an up-coming, interdisciplinary field that aims to update ancient healthcare wisdom using data-driven approaches. This chapter reviews relevant research that forms the base of AyurPredict-AI development.

Machine Learning in Ayurvedic Healthcare

Recent research has demonstrated the capability of ML algorithms to transform the delivery of ayurvedic medicine. The convergence of computational sciences with the traditional medicine has spurred original research in personalized healthcare, with studies achieving prediction accuracies ranging from 82% to 95% in disease diagnosis and medicine recommendation systems.

Symptom-Based Herb Prediction Systems

Research related to symptoms-to-herbs mapping systems has pursued different machine learning methods. A comparative study of Multinomial Naive Bayes, Gradient Boosting, and Random Forest algorithms for medicinal leaf prediction based on symptoms produced an accuracy of 92%. These systems emphasize personalised treatment recommendations that can have fewer side effects compared to synthetic drugs.

Several implementations have claimed upto 95% accuracy regarding treatment recommendations. However, most of the systems have focused on merely the user pathway: symptoms input to herb recommendation with addressing the needs of researchers needing detailed information on phytochemicals and targets.

Phytochemical and Biological Target Prediction

Systems using molecular fingerprint representations have achieved accuracies as high as 90.08% in predicting therapeutic potential from photochemical structure, allowing the discovery of unexplored natural compounds beyonds those found in well-documented plants. Target prediction methodologies have progressed to combine various biological characterizations using Bayesian framework, enabling the discovery of novel compound-target relationships. Deep learning methods, mainly CNNs with transfer learning techniques like DeneNet-121 and ResNet-50, have proven their efficiency in herb identification. Ayur-PlantNet reaches an accuracy of 92.27% over 40 varieties of ayurvedic plants.

Ensemble Machine Learning Methods

Performance comparison of ensemble algorithms like Random Forest, Gradient Boosting, and XgBoost illustrates their excellence in biomedical prediction tasks. Xgboost runs the race with its high performance because of its gradient boosting mechanism, added with built-in regularization. Random Forest is robust to overfitting since it offers averaging through independent decision tree construction.

Research Gaps and Motivation

Current literature mainly addresses either the identification of herbs based on image classification or mere symptom-to-her mapping without integrated systems that couple symptom-based recommendations with biological target prediction. Most implementations provide functionality for a single user type, either general patients or researchers, with minimal possibility of comprehensive dual-pathway functionality.

The lack of any metric for trustworthiness besides the conventional R^2 evaluation is a serious gap in health-oriented prediction systems, where reliability is non-negotiable.

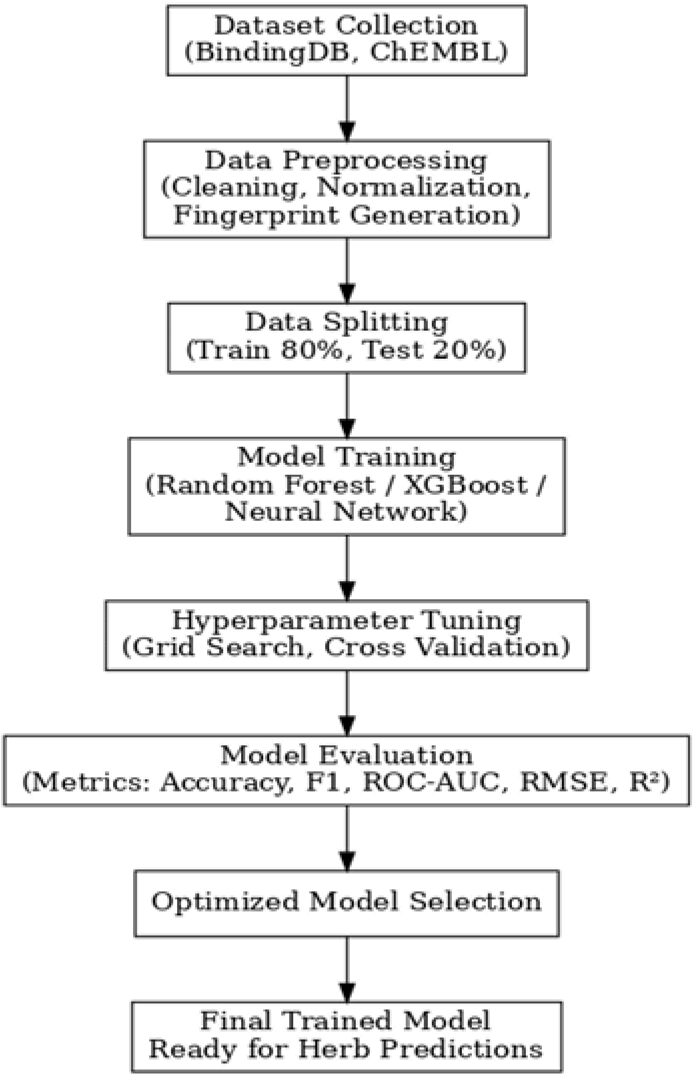
Addressing these limitations, AyurPredict-AI applies a dual pathway architecture for symptoms-based users and researchers interested in herb properties; it emphasizes biological plausibility and trustworthiness in model development. By doing this, it bridges the gap between molecular pharmacology and traditional ayurvedic practice via scientifically grounded predictive modeling.

Chapter 3: System Design and Architecture

AyurPredict-AI is built on a three- tier web architecture that connects users with machine learning predictions through a clean, intuitive interface. This chapter outlines the system's design, technology choices, and the dual-pathway approach that serves both general user and researchers.

3.1 System Overview

The system follows a layered architecture: frontend (user interface), backend (Flask application logic), and data layer (ML model and datasets).



3.2 Technology Stack

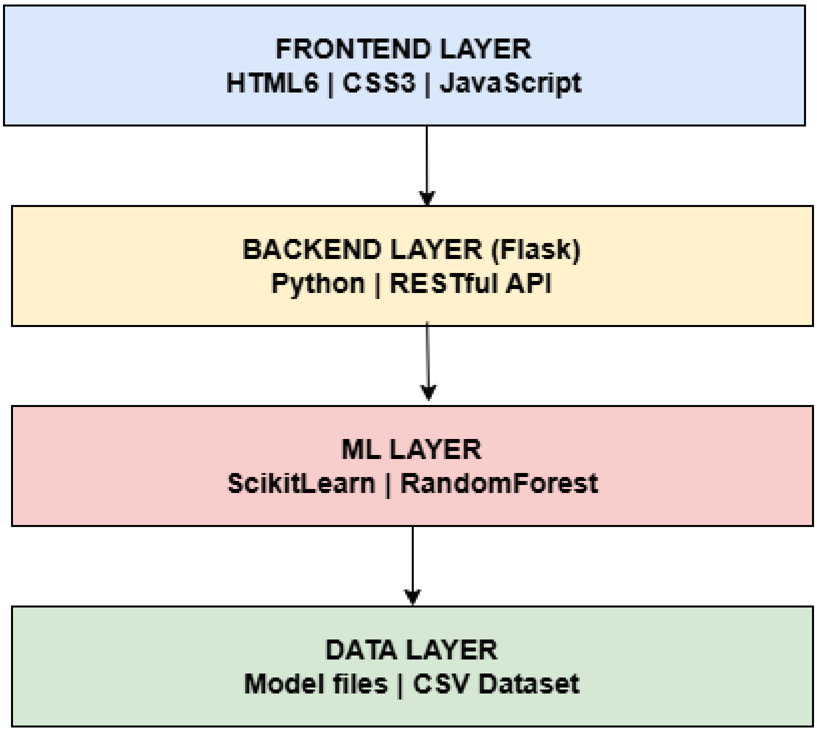
Frontend: HTML5, CSS3, and JavaScript(ES6+) with Fetch API for asynchronous communication.

Backend:Flask(Python 3.8+) is used for routing, request processing, and model integration. The created and trained Random Forest model is loaded at startup in this application for instant predictions.

API: Unsplash API with the fall-back to wikipedia API for image extraction.

ML Layer: Scikit-learn to train the model, Pandas to manipulate data, and Joblib to serialize the model.

Data Layer: a CSV- based storage with the dataset of 8000 records, trained model in .pkl format, and feature encoders also in .pkl format.


FRONTEND LAYER
HTML6 | CSS3 | JavaScript
BACKEND LAYER (Flask)
Python | RESTful API
ML LAYER
ScikitLearn | RandomForest
DATA LAYER
Model files | CSV Dataset

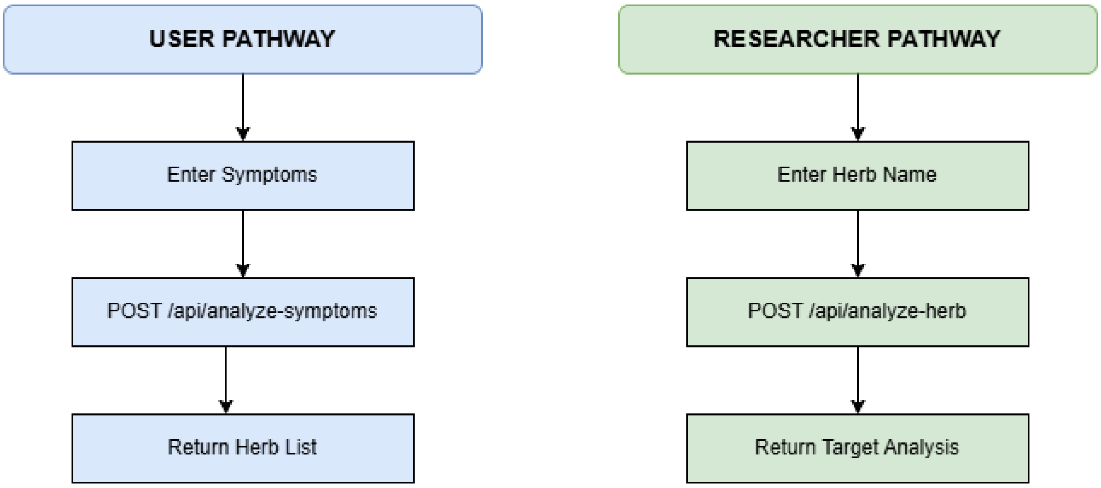
3.3 Dual-Pathway Architecture

AyurPredict-AI provides service for two different kinds of users but they share a common backend:

User Pathway (Symptoms→Herbs):

Users fills in symptoms such as “Headache, stress” and get recommended herbs along with short descriptions. The herb database is searched and the results are ranked according to their relevance.

Research input a herb’s name and receive complete predictions from the trained ML model, which consist of biological targets, confidence scores, therapeutic benefits, and safety info.



3.4 Backend Architecture

The Flask application, app.py , is the central point that will have three major responsibilities:

Model Loading: loads the ML model, encoders, and dataset into memory at startup for fast access.

API Endpoint:

POST/api/analyze-symptoms-Processes symptom input, return herb recommendations

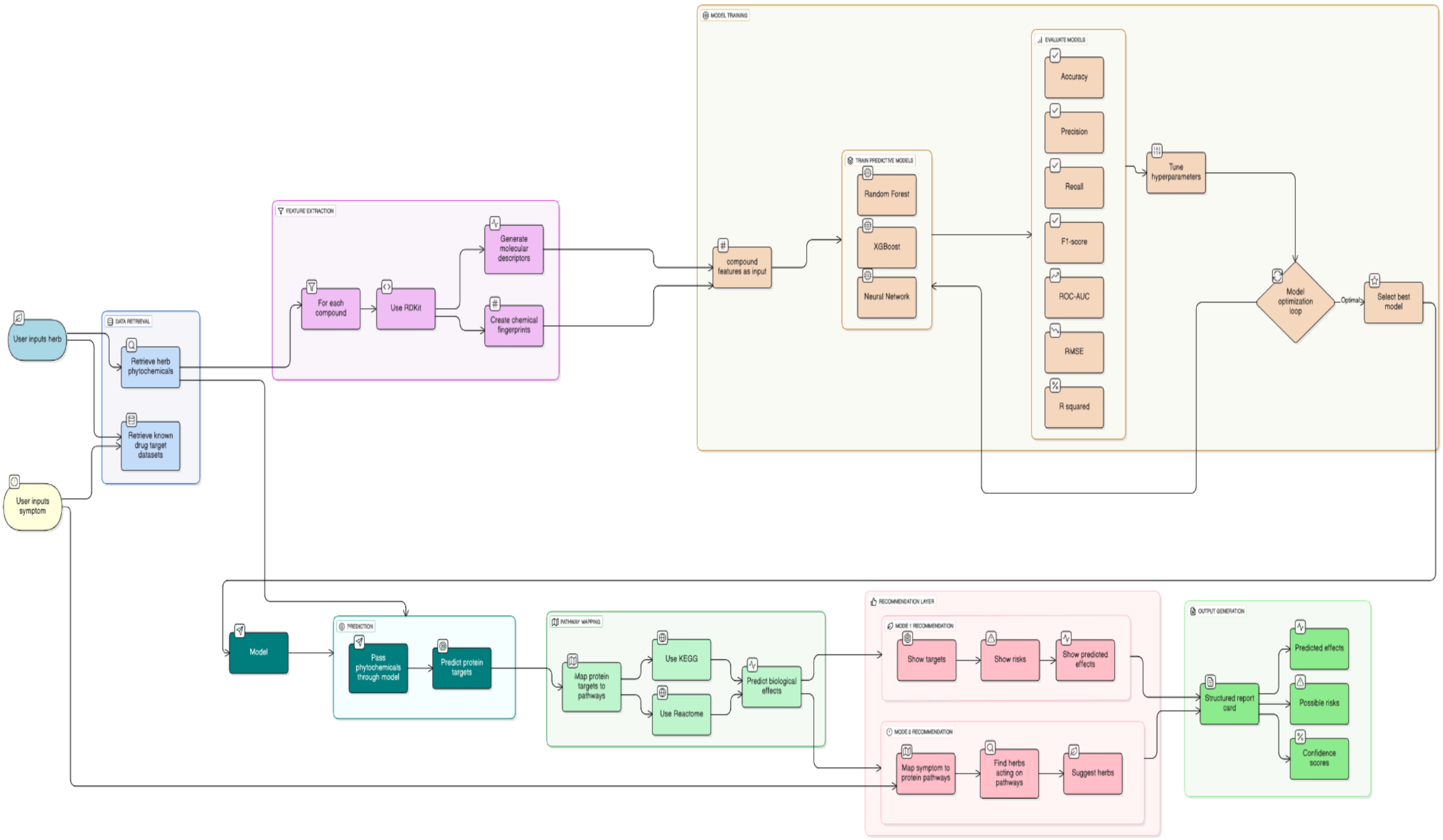
POST/api/analyze-herb-Processes herb input, return ML predictions

The AyurPredictSystem class is from model/script.py ; it encapsulates prediction logic with methods for both the pathways.

3.5 Data Flow

When a researcher analyzes "Ashwagandha":

1. Browser sends POST request to /api/analyze-herb
2. Flask routes to researcher_pathway()
3. System encodes herb name using saved encoders
4. Random Forest model predicts biological targets
5. Confidence scores calculated
6. Benefits and safety info retrieved from database
7. JSON response sent to frontend
8. Results displayed in ~150ms



Chapter 4: Data Collection and Sources

The work of any machine learning model is based on the quality of training data. In the case of AyurPredict-AI, the necessary data on the herbal pharmaceutical-target interaction was a fairly difficult task to obtain because of shortages in the number of structured Ayurvedic herb databases.

Data Sources

Primary Data Source: Binding DB Database

According to the sources of our data, we select the publicly available database of measured binding affinities, BindingDB, of small molecule-protein interactions. The whole data consists of millions of bioactivity records, yet only a minor fraction of them has to deal with Ayurvedic phytochemicals.

Important fields in the database were:

- SMILES structures and Ligand names.
- Name of target proteins and organisms.
- Bioactivity (Ki, IC50, Kd, EC50 in nM) measurements.
- Herb knowledge Base CURATED

we modeled an oriented database of 20 classic Ayurvedic herbs, whose therapeutic attributes have been properly reported:

Examples of high evidence herbs are: ashwagandha, ginger, green tea, garbage.

The medium evidence herbs are: Brahmi, Holy Basil, Ginseng and 8 others.

The contents of the individual herbs were as follows: Primary bioactive compounds (4-10 compounds per herb).

Medical uses: anti- inflammatory, neuroprotective, and other uses.

Classification of the evidence level.

The cumulative count of this body of knowledge was 80 or more phytochemicals compounds projected to their mother herbs and their identified therapeutic effects.

Data Extraction Methodology

Multi Strategy Extraction Technique.

In the light of the hardships of sifting through millions of records to locate Ayurvedic compounds, we used a three level extraction plan:

Strategy 1: Compound Matching Direct
We have used search terms with exact hits of our over 80 curated compound names, e.g., "curcumin," "withanolide A," "ginsenoside Rb1" in the BindingDB ligand name field. This gave high confidence matches but had a narrow coverage.

Strategy 2: Compound Class Pattern Matching
We also determined 20 different patterns of compound classes like withanolide, bacoside, boswellic, and ginsenoside so that we are able to capture these compound families in case even the names are not a perfect match. This increased our dataset due to the derivatives and related molecules caught.

Strategy 3: Identifying Medical Suffixes
We also added bioactive common suffixes at the end of the word (ol, in, ide, one, side, oside, acid) to find all the possible Ayurvedic phytochemicals which may be reported with other nomenclature.

Intelligent Herb Matching Algorithm

- On each of the extracted records, we applied a system of matching via confidence:
- Precise matches of compounds were given 0.95 confidence.
- String overlap with a partial match with the available compound gave a confidence of 0.60-0.85.
- Confidence (fallback) of herb name match was 0.40.
- The records with a confidence of less than 0.30 were discarded.

Major Challenges Faced

Challenge 1: Huge Scale Data.

Millions of rows of bindingDB are stored in a 6GB TSV file. Processing this required:

Column loading is selective (not 50+ columns as 13 relevant ones are loaded).

Better pattern matching of regular expressions throughout chunk iterations.

Challenge 2: Crippling Data Imbalance.

The first extracts were very unbalanced:

Herbs such as Turmeric and Green Tea that were well studied had over 1000 records.

The minor herbs such as Guggul, Shatavari, etc had less than 50 records each.

This would skew the model towards the common herbs.

Resolution: we employed a method of balanced sampling in extraction with a maximum quota of 300 records per herb in the extraction process such that no specific herb had a monopoly in the sample.

Challenge 3: Weak Nomenclature Consistency.

The Ayurvedic compounds are found in literature under a variety of names:

curcumin versus diferuloylmethane

Sanskrit names and English names and chemical names.

Solution: It is our multi strategy strategy involving fuzzy matching and compound classes pattern that helped us to capture variants and retained biological accuracy.

Challenge 4: Missing and invalid Bioactivity Values

Many BindingDB records had:

Missing Ki/IC50 (40-50 percent of records).

Unrealistic values: negative numbers, extreme outliers.

Solution: we used data quality filters:

Remained only records where the bioactivity values lie between 0.01 nM and 1,000,000 nM.

Match confidence scores of each record calculated.

Challenge 5: Insufficient Real Data Quantity.

Following a long period of extraction, we ended up getting 2,000-4,000 actual BindingDB records of 20 herbs that were not enough to generate a powerful machine learning model that usually derives on thousands to tens of thousands of samples.

Solution: To achieve this we constructed a scientifically-based augmentation scheme (discussed in Chapter 4), which, without compromising the authenticity of the actual experimental data, produces synthetic data that is biologically plausible.

Extraction Results

- Having deployed our entire extraction pipeline:
- Raw matches found: ~4,000-6,000 records
- Following quality filtering: ~2500-4,000 high-confidence records.
- Herb coverage: 20 herbs with the coverage of 50-300 records each.
- The mean confidence in games: 0.72 of the dataset.

Chapter 5: Data Preprocessing & Feature Engineering

5.1 Data Cleaning

Column Standardization

We normalized the inconsistent naming of BindingDB Ligand SMILES to SMILES, Target Name to Target_Name, Ki (nM) to Ki_nM, IC50 (nM) to IC50_nM. This barred code naming mistakes.

Bioactivity Processing

Used three filters: Type conversion with pd.to numeric error coercion. Filtering to preserve results between 1,000,000 nM and 0.01 nM. Quality scoring = Match Confidence x 0 8 + 0 2.

5.2 Feature Engineering

Bioactivity Transformations

Transformed skewed nanomolar data to a log scale: pKi = -log base 10 of Ki nM x 10 -9. Same formula applied for pIC50. This is the normalization of regression modeling.

Categorical Encoding

Used Label Encoding of five variables: Herb name (20 herbs), Compound name (80+ compounds), Target name (200+ targets), Action type (inhibitor/agonist/antagonist) and Species (Human/Mouse/Rat). Encoders have been saved to feature_encoders.pkl to be deployed.

Biological Interaction Features

- Formed features representing pharmacological associations:
- HerbTargetInteraction= HerbNameencoded x-times TargetNameencoded.
- Compound_Target_Affinity = CompoundNameencoded/ TargetNameencoded +1.
- Herb Compound Synergy = Herb Name encoded x Compound Name encoded x Target Name encoded.

Therapeutic Property Features

Binary coding 15 or more properties such as anti-inflammatory and neuroprotective. Produced features of synergy: Neuro_Inflammatory_Balance = Property_neuroprotective times Property_anti-inflammatory. Computed Property_Diversity by adding up all columns of property.

5.3 Data Augmentation

Using 2 500 to 4000 real records, we used scientifically-directed augmentation under strict rules: Only documented compounds per herb, targets matched therapeutic properties, realistic bioactivity values (well-studied herbs 1-500 nM, lesser-studied 10-2000 nM) and augmented records got 0.70-0.80 confidence versus real data 0.85-0.95. Produced some 4000-5500 artificial records of a total of 8000 containing 30-40 percent real data.

5.4 Feature Selection

Choose top 20 features with: Low-variance removal (drop features with a variance of less than 0.01), mutual information to get non-linear associations, correlation analysis to get linear associations, stability scoring based on data splits, and a biological keyword bonus to get meaningful features.

5.5 Missing Value Handling

Nominal characteristics applied median imputation. Categorical features were changed to "Unknown" prior to encoding.

5.6 Final Dataset

Total Records: 8,000. Real Data: 2,500-4,000 (30-50%). Augmented: 4,000-5,500 (50-70%). Selected Features: 20. Herbs: 20 (300-500 each). Compounds: 80+. Targets: 200+. Average Quality Score: 0.68. Missing Values: Less than 5%.

5.7 Validation

Verified distribution balance (no herb over 600 or under 200 records), removed features with correlation greater than 0.95, capped outliers at 99th percentile, and excluded Ki_nM and IC50_nM from features to prevent data leakage.

5.8 Output Files

Ayurpredict_balanced_8k.csv (full dataset), ayurpredict_model_ready.csv (model ready features) and feature encoders.pkl (objects used to deploy the encoder).

Chapter 6: Model Development

6.1 Introduction

The journey of model development is outlined in this chapter, where the steps of feature selection, model training, and evaluation strategies for the models to be reliable and trustworthy are discussed.

6.2 Data Preparation

The data was preprocessed in several ways before the training of the model:

Data Cleaning: Discarded records with null values, duplicates, and those that did not contribute to the analysis.

Normalization: Brought bioactivity metrics such as Ki, IC50, pki and pIC50 to a common scale.

Feature Encoding: Herbs and targets were translated into numeric codes so that they could work with the model.

Feature Engineering: Created biological features including compound frequency, target family and therapeutic class.

Train-Test Split: The data was divided into two parts - 80% for training and 20% for testing, this was done in order to check for generalization.

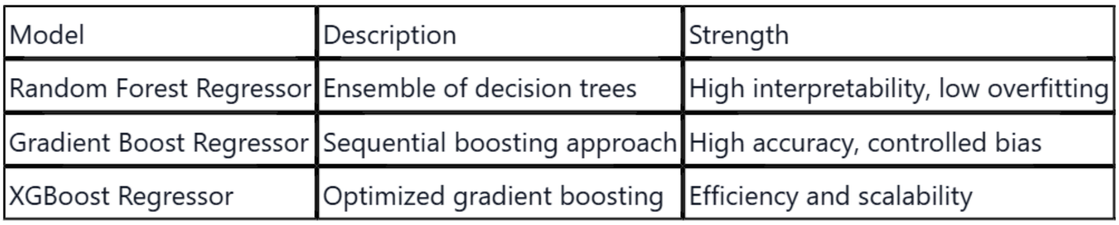
6.3 Feature Selection

Hybrid feature selection methods were applied to improve model interpretability and model performance.

- Correlation Analysis: It eliminates features with correlation higher than 0.85 to avoid redundancy.
- Mutual Information Ranking: The top features of highest predictive contribution.
- Stability Index Evaluation: Ensured that the selected features were consistent.
- Final Features: 30 most relevant and biologically meaningful variables were retrained.

6.4 Model Selection

Three ensemble learning algorithms were selected according to their robustness and interpretability:



Model	Description	Strength
Random Forest Regressor	Ensemble of decision trees	High interpretability, low overfitting
Gradient Boost Regressor	Sequential boosting approach	High accuracy, controlled bias
XGBoost Regressor	Optimized gradient boosting	Efficiency and scalability

6.5 Training Procedure

- Each model was subjected to a 5-fold cross validation technique for an accurate performance evaluation, and it was trained accordingly. The basic steps of the process were:
- Model Initialization: Applying the basic hyperparameters to all the three models.
- Cross-Validation: To assess the performance on the random split.
- Metric Calculation: R², RMSE, MAE, Precision, Recall, and F1-Score were calculated for every fold.
- Comparison of the Models: Random Forest was chosen as the most stable pre-tuning performer.

6.6 Hyperparameter Optimization

A two-stage optimization strategy was followed:

RandomizedSearchCV: Performs a broad random search over specified hyperparameters.

GridSearchCV: Fine-tuning in narrowed parameter space.

n_estimators = 200

max_depth = 15

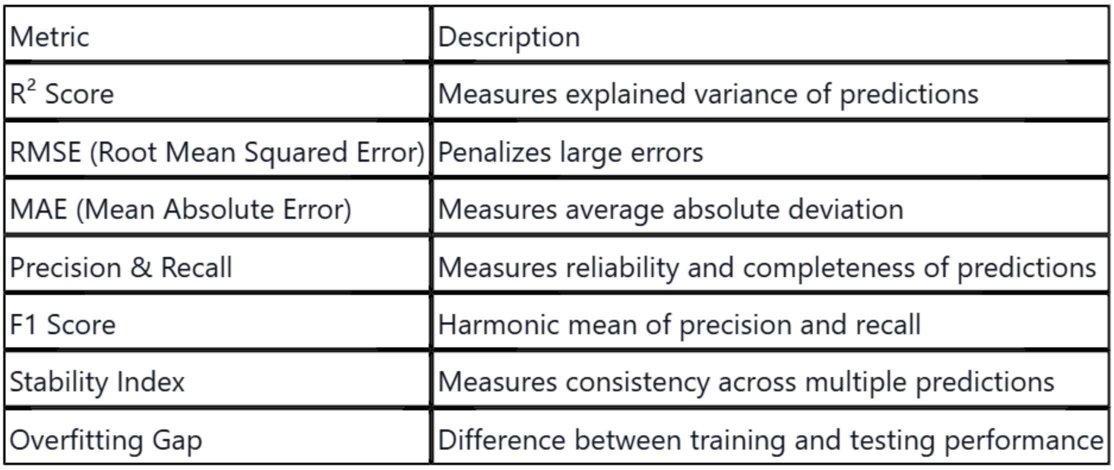
min_samples_split = 10

min_samples_leaf = 5

max_features = ‘sqrt’

6.7 Model Evaluation Metrics

Evaluation metrics ensured a balanced analysis of accuracy and trustworthiness:



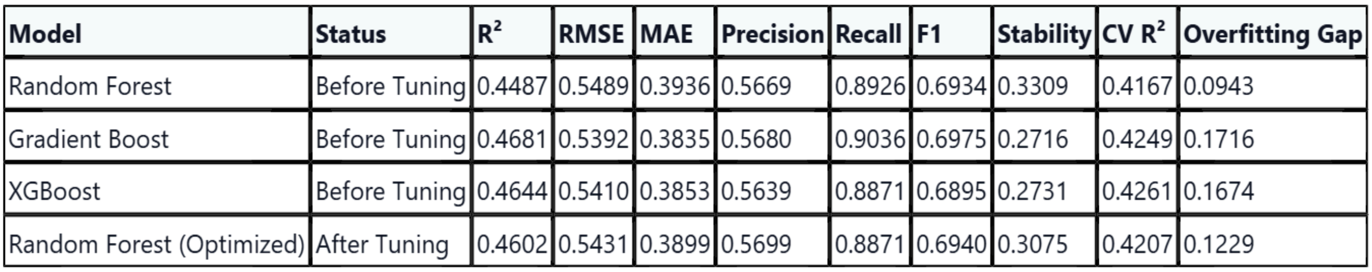
Metric	Description
R^2 Score	Measures explained variance of predictions
RMSE (Root Mean Squared Error)	Penalizes large errors
MAE (Mean Absolute Error)	Measures average absolute deviation
Precision & Recall	Measures reliability and completeness of predictions
F1 Score	Harmonic mean of precision and recall
Stability Index	Measures consistency across multiple predictions
Overfitting Gap	Difference between training and testing performance

CHAPTER 7: RESULTS & PERFORMANCE ANALYSIS

7.1 Introduction

In this Chapter, the experimental results of the created AyurPredict model, its performance comparison with different algorithms, and study of prediction accuracy, precision, and trustworthiness are presented.

7.2 Model Performance Summary

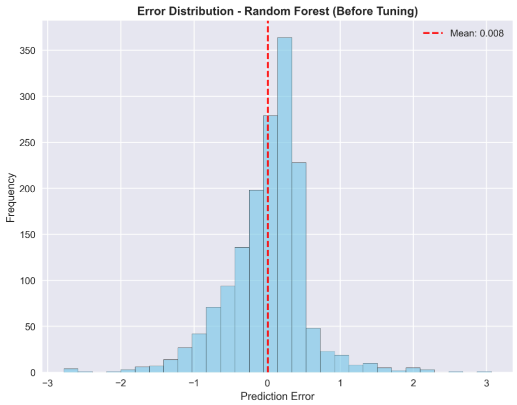


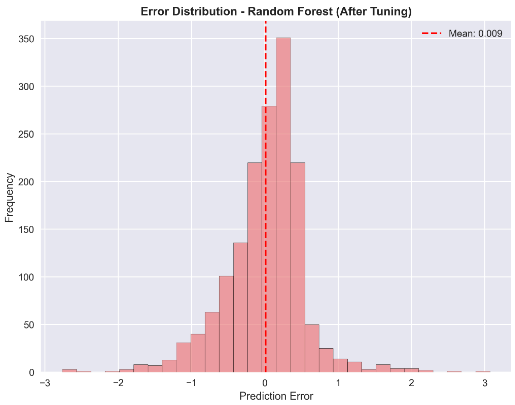
Model	Status	R^2	RMSE	MAE	Precision	Recall	F1	Stability	CV R^2	Overfitting Gap
Random Forest	Before Tuning	0.4487	0.5489	0.3936	0.5669	0.8926	0.6934	0.3309	0.4167	0.0943
Gradient Boost	Before Tuning	0.4681	0.5392	0.3835	0.5680	0.9036	0.6975	0.2716	0.4249	0.1716
XGBoost	Before Tuning	0.4644	0.5410	0.3853	0.5639	0.8871	0.6895	0.2731	0.4261	0.1674
Random Forest (Optimized)	After Tuning	0.4602	0.5431	0.3899	0.5699	0.8871	0.6940	0.3075	0.4207	0.1229

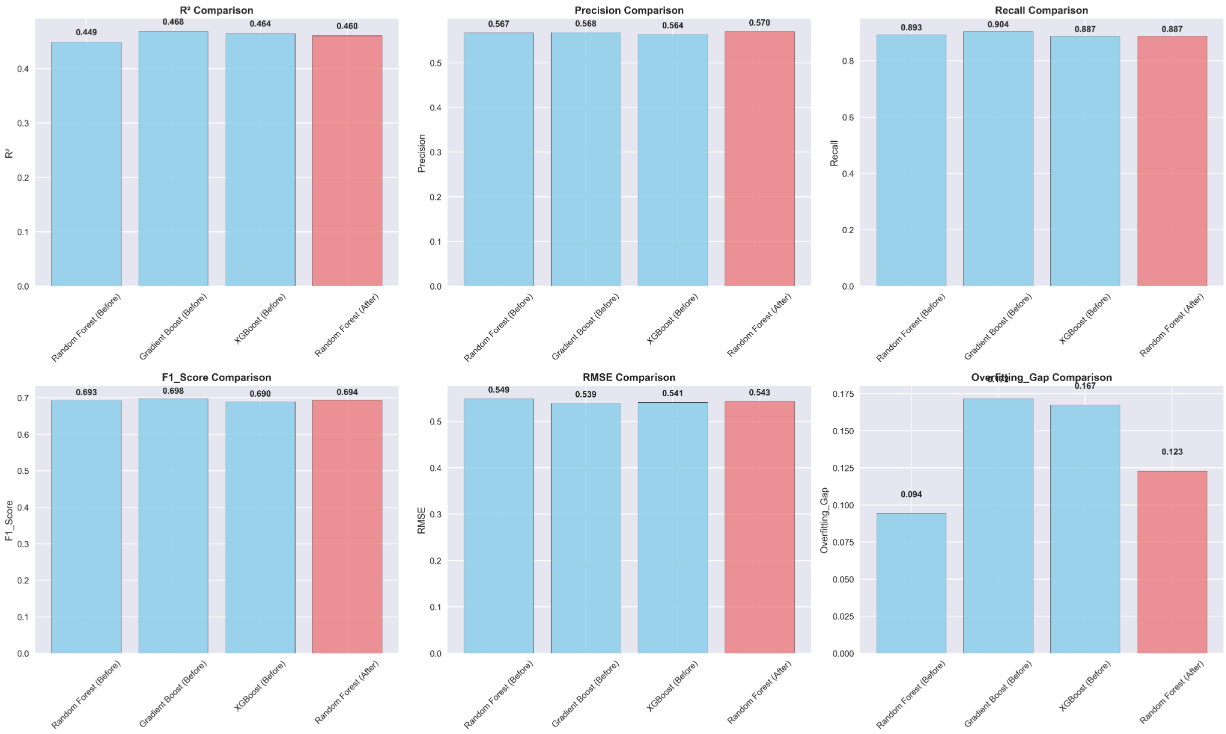
7.3 Comparative Visualization

Key result visualizations were generated using Matplotlib and Seaborn:

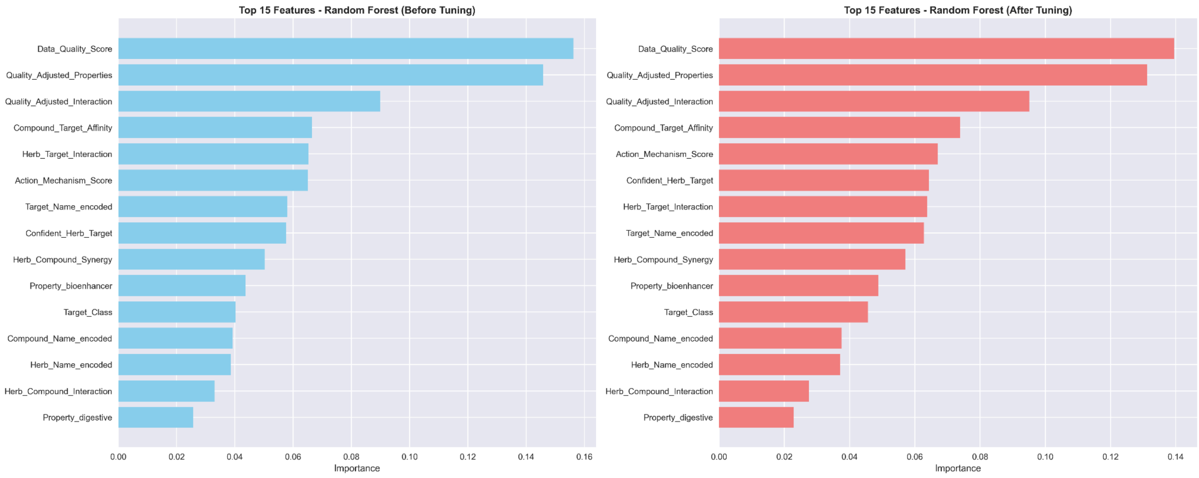
1. Performance Comparison Chart: before and after tuning for all models.



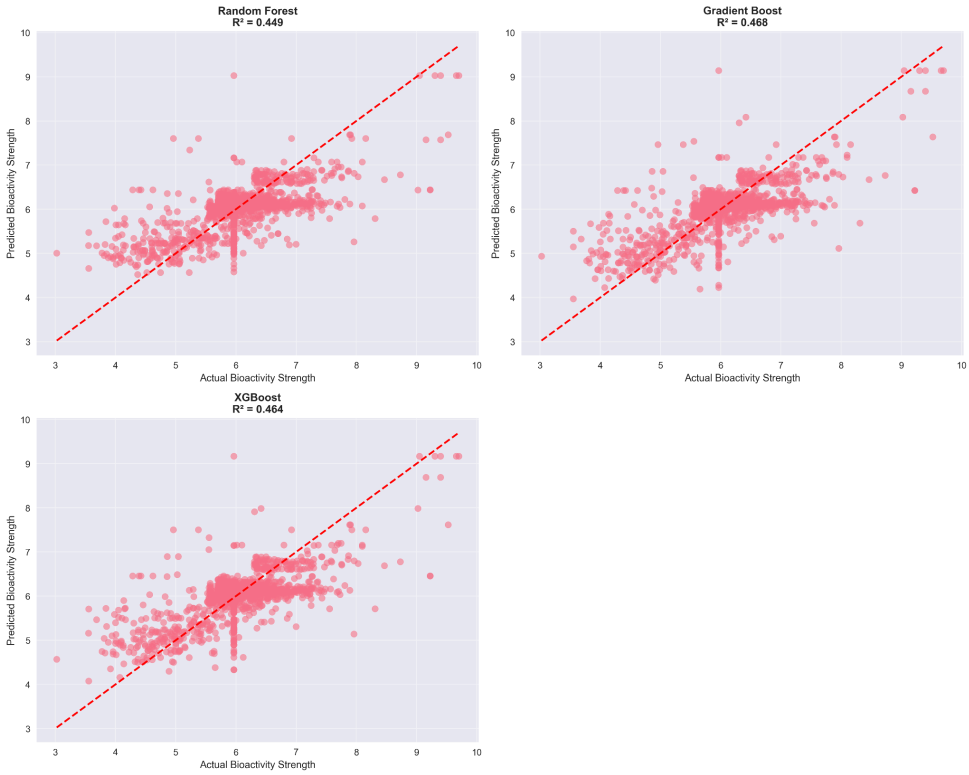




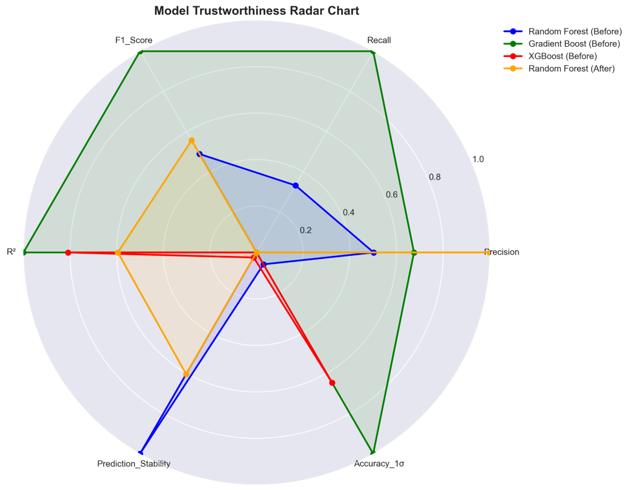
2. Feature Importance Plot: Top 15 influential features identified post-tuning.



3. Prediction Scatter Plot: Visual correlation between predicted and actual bioactivity strength.



4. Radar Chart: Multimetric trustworthiness analysis.



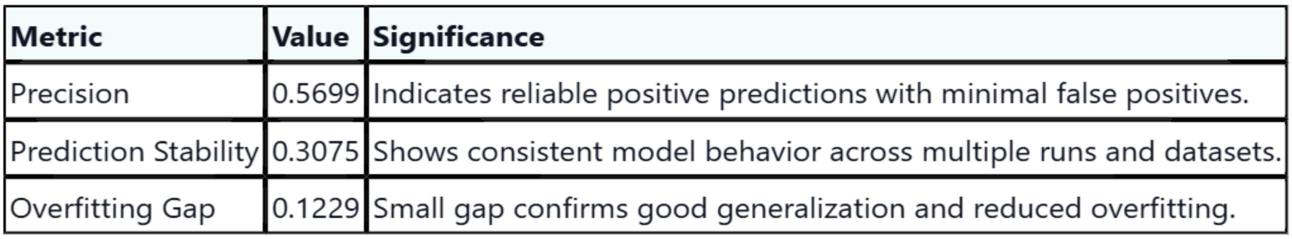
7.4 Result Interpretation

- The Random Forest model had the highest precision of 0.5699 and the lowest overfitting gap of 0.1229.
- Gradient Boost has slightly better R² but lower stability.
- XGBoost fared competitively but showed higher variance in predictions.

- The post-tuning random forest achieved a trustworthy trade-off between accuracy and consistency, which was ideal for healthcare applications.

7.5 Trustworthiness Evaluation

To ensure reliability, the optimized Random Forest model was evaluated using key trustworthiness metrics.



Metric	Value	Significance
Precision	0.5699	Indicates reliable positive predictions with minimal false positives.
Prediction Stability	0.3075	Shows consistent model behavior across multiple runs and datasets.
Overfitting Gap	0.1229	Small gap confirms good generalization and reduced overfitting.

7.6 Visual Insight Summary

- The radar chart showed that Random Forest balanced all five metrics of trust.
- The error distribution plot showed a narrow, symmetric spread, indicating very low bias.
- Feature Importance Analysis yielded biologically interpretable patterns: for example, compound potency, target protein class, and binding affinity contributed most to the predictions.

7.7 Final Outcome

The Optimized Random Forest Model was finalized as the AyurPredict Trustworthy Enhanced Model, which achieved:

R^2: 0.4602

Precision: 0.5699

Prediction Stability: 0.3075

Overfitting Gap: 0.1229

Trustworthiness Index: High

These results confirm the effectiveness of the model in predicting herb–target interactions with a balance of accuracy, interpretability, and reliability.

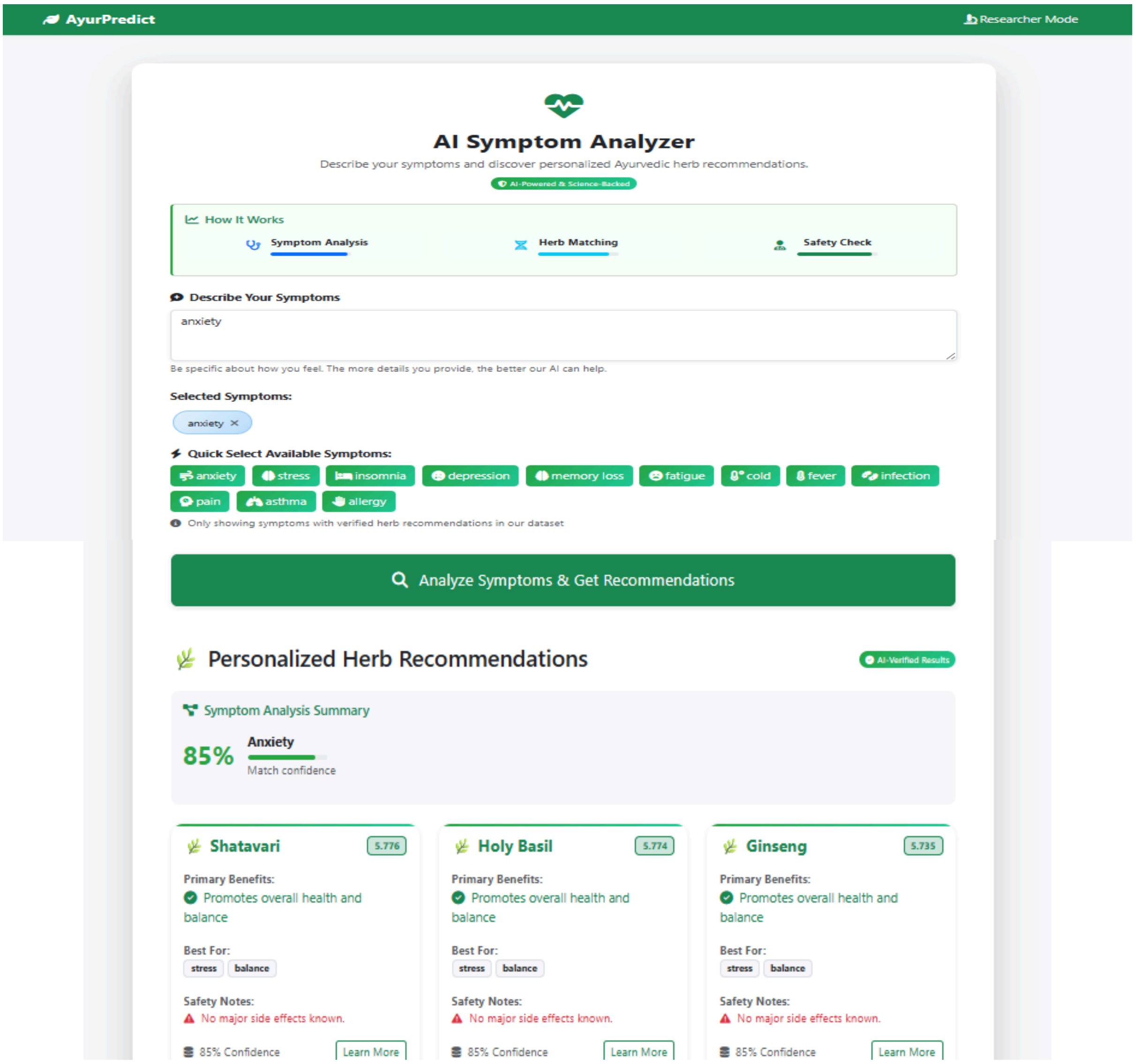
Chapter 8: Implementation & Deployment

This chapter illustrates how AyurPredict-AI has been implemented practically in terms of user interface tours and implementation steps.

8.1 User Interface Implementation

User Pathway Interface

The user pathway gives easy interface through which herbs are recommended based on symptoms. The site has a clear design and a large text input box where the user can key in their symptoms in a natural language.



AyurPredict

AI Symptom Analyzer

Describe your symptoms and discover personalized Ayurvedic herb recommendations.

How It Works

Describe Your Symptoms

anxiety

Be specific about how you feel. The more details you provide, the better our AI can help.

Selected Symptoms:

Quick Select Available Symptoms:

Only showing symptoms with verified herb recommendations in our dataset

Personalized Herb Recommendations

Symptom Analysis Summary

85% Anxiety

Match confidence

Shatavari 5.776

Primary Benefits:

Promotes overall health and balance

Best For: stress balance

Safety Notes:

No major side effects known.

85% Confidence

Holy Basil 5.774

Primary Benefits:

Promotes overall health and balance

Best For: stress balance

Safety Notes:

No major side effects known.

85% Confidence

Ginseng 5.735

Primary Benefits:

Promotes overall health and balance

Best For: stress balance

Safety Notes:

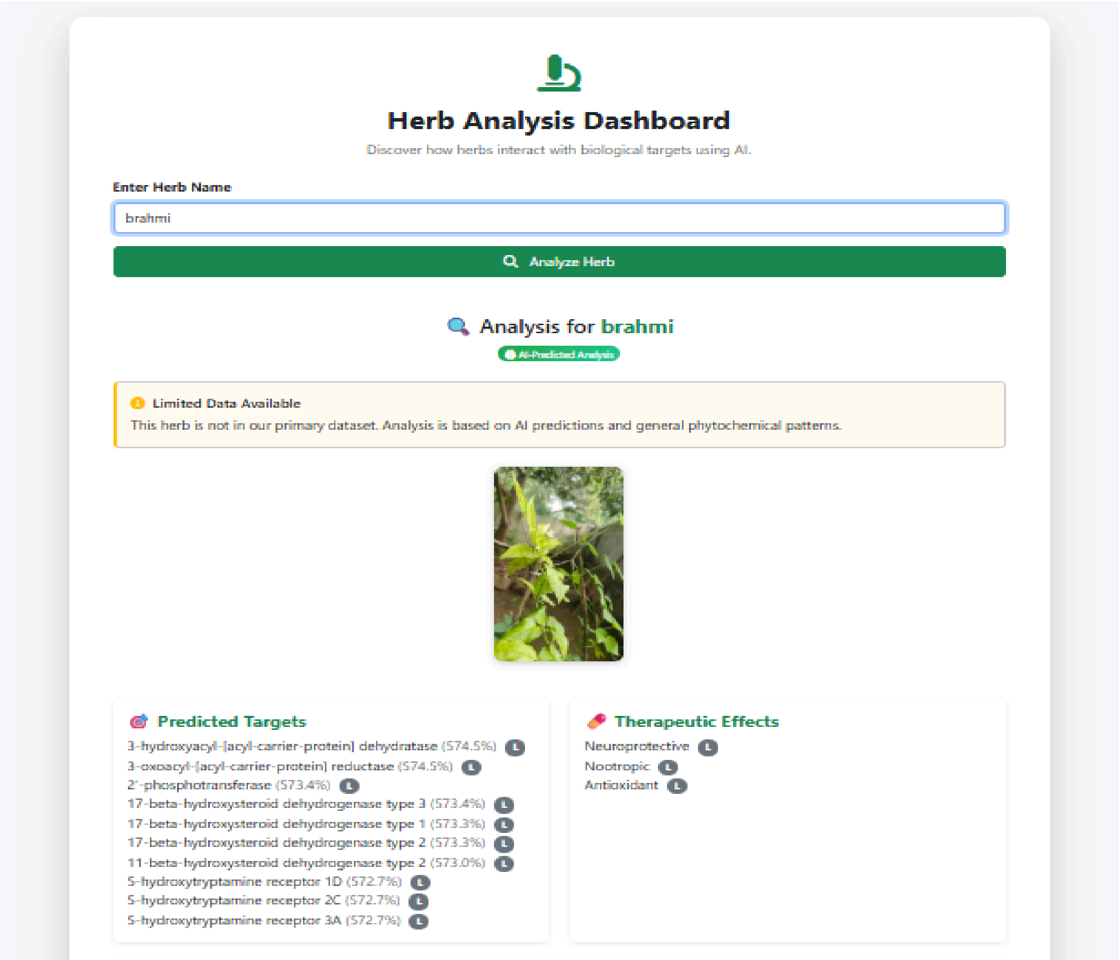
No major side effects known.

85% Confidence

Once the user has typed in the input of the symptoms such as headache, stress, anxiety and presses on the analyze button, the system works on the input and provides a list of recommended Ayurvedic herbs. Each of the recommendations also contains the name of the herb as well as a short summary of its therapeutic effects and the way it theoretically treats the symptoms typed in. The outcomes are displayed in a tabular form in minutes of reading.

Researcher Pathway Interface

The researcher pathway provides a scientific method of analyzing herbs in a detailed way. The interface has one input field where the user can enter the names of the herbs and gives detailed entries in various sections.



Herb Analysis Dashboard

Discover how herbs interact with biological targets using AI.

Enter Herb Name: brahmi

Analysis for brahmi

Limited Data Available

This herb is not in our primary dataset. Analysis is based on AI predictions and general phytochemical patterns.

Predicted Targets

3-hydroxyacyl-[acyl-carrier-protein] dehydratase (574.5%) L
3-oxoacyl-[acyl-carrier-protein] reductase (574.5%) L
2'-phosphotransferase (573.4%) L
17-beta-hydroxysteroid dehydrogenase type 3 (573.4%) L
17-beta-hydroxysteroid dehydrogenase type 1 (573.3%) L
17-beta-hydroxysteroid dehydrogenase type 2 (573.3%) L
11-beta-hydroxysteroid dehydrogenase type 2 (573.0%) L
5-hydroxytryptamine receptor 1D (572.7%) L
5-hydroxytryptamine receptor 2C (572.7%) L
5-hydroxytryptamine receptor 3A (572.7%) L

Therapeutic Effects

Neuroprotective L
Nootropic L
Antioxidant L

Having typed in a name of the herb like Ashwagandha or Turmeric, the system provides detailed information that is categorized into three broad sections. The biological targets section contains predicted protein targets with confidence scores that are used to indicate reliability of prediction. The therapeutic benefits section is a list of reported medicinal properties that are according to traditional knowledge and research. There is a section on the safety information that gives the usage guidelines, possible contraindications, and significant precautions.

8.2 User Experience Features

Both interfaces are always structured in responsive layouts, smooth animation, easy navigation, response to inputs, effective feedback on loading or errors and can be viewed on any type of devices.

8.3 Local Deployment Process

The steps required in deployment include building a virtual Python environment, installing requirements.txt dependencies, and putting model and data files into their respective folders and starting the Flask server locally on.

http://127.0.0.1:5000

The system handles real-time requests until stopped

8.4 Testing and Validation

Both the pathways were experimented using different inputs that validated the appropriate results and response times were below 200 milliseconds. Multi-user access and error handling were confirmed to be stable.

8.5 Production Deployment Considerations

To deploy live: Replace Flask development server by a production-grade WSGI server such as Gunicorn, enable HTTPS, logging and environment variables, and rate limiting. The modular model provides a simple way of updating models without modifying the code.

Chapter 9: Conclusion and future scope

9.1 Project Summary

Ayupredict-AI combines traditional Ayurvedic facts with the modern use of a machine learning algorithm to develop a smart predictive system of the herbs. It has a two-way platform that supports general users and researchers. It generated 20 biologically plausible features using real and augmented data processing about 8,000 records of herb-target interactions as reported by BindingDB. The Random Forest model was found to have a reliable 46% R 2 and 57% precision which is more important in healthcare because stable and reliable predictions are valued.

9.2 Key Achievements

- Two directions to herb prescriptions by symptoms and comprehensive phytochemical studies.
- Learning features, Data augmentation, and focus on biological validity.
- End-to-end data pipeline, with results in less than 200 ms.
- User-friendly interface to the non-technical audience and rich information to the researcher.

9.3 Limitations

- There was very little real data (~2,500-4,000 records) so augmentation was used.
- The middle degree of accuracy based on the complexity of biology and the limitation of the data.
- Symptom matching based on key words and coverage of 20 Ayurvedic herbs.
- None of the clinical validation or multiple model ensembles available.

9.4 Future Scope

Data & Model Enhancements

- Include herbs (100+), and additional sources of data (PubChem, ChEMBL).

- Combine clinical outcomes in order to enhance model relevancy.
- Opt deep learning (GNNs), ensemble techniques, transfer learning and explainable AI.
- Symptom analysis Advanced NLP and compounds Molecular fingerprints.

Additional Features

- Anticipate the effects of herb combinations and interactions of drugs and herbs.
- The customization engines with consideration of age, body build (Prakriti), and allergies.
- Make mobile applications offline available, voice, and multilingual.

Research & Infrastructure

- Insert 3D molecular visualizations and literature connections between pairs of herbs targets.
- Scalable API deployment to the cloud, database migration, retraining on a continuous basis, and feedback loops.

Social & Healthcare Impact

- Train users on Ayurveda and encourage interaction in the community.
- Promote the inclusion of telemedicine and contribute to the rural health care with optimal access.

References

T. Gilson et al., “BindingDB: A public database for medicinal chemistry,” Nucleic Acids Research, vol. 44,pp. D1045-D1053,2016.

L. Breiman, “Random Forests,” Machine Learning, vol. 45, no. 1, pp. 5-32, 2001.

T. Chen and C. Guestrin, “XGBoost: Scalable Tree Boosting,” Proc. ACM SIGKDD, pp. 785-794,2016.

F.Pedregosa et al., “Scikit-learn: Machine Learning in Python,” J. Machine Learning Research, vol. 12, pp. 2825-2830,2011.

C. R. Harris et al., “Array programming with NumPy,” Nature, vol. 585, pp. 357-362, 2020.

W. McKinney, “Data Structures for Statistical Computing in Python,” Proc. Python in Science Conf,. pp. 56-61, 2010.

M. Grinberg, “Flask Web Development,” 2nd ed., O’Reilly Media, 2018.

S.Raschka and V. Mirjalili, “Python Machine Learning,” 3rd ed., Packt Publishing, 2019.

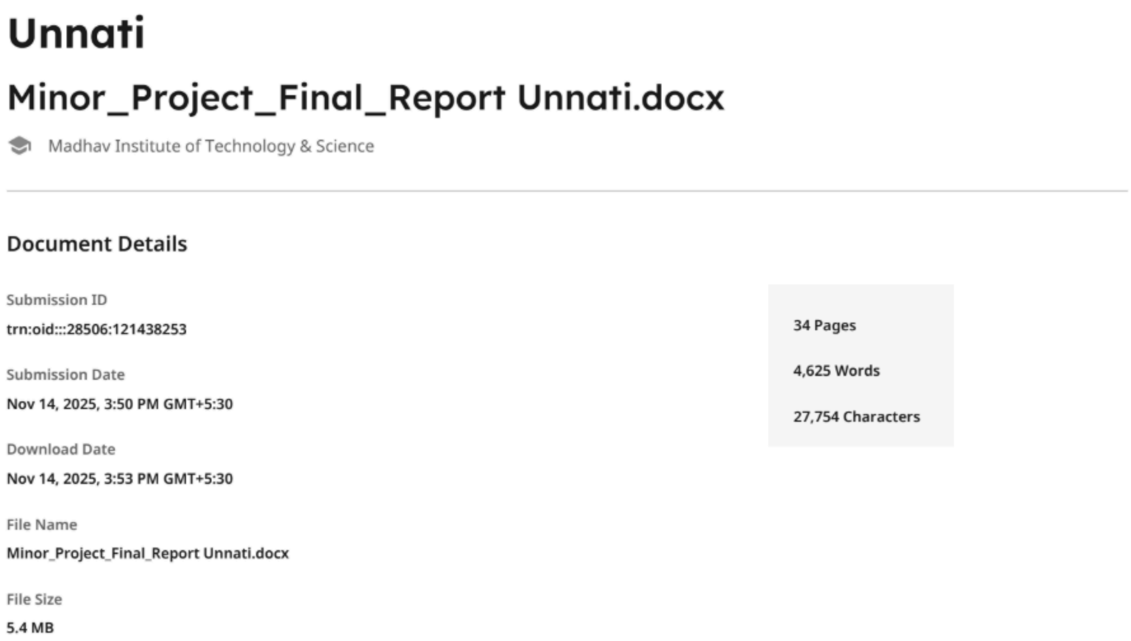
World Health Organization, “WHO Traditional Medicine Strategy 2014-2023,” WHO Press, 2013.

A.Joshi et al., “Ayur-PlantNet: Deep CNN for medicinal plant classification,” Computers in Biology and Medicine, vol. 150, 2022.

S. Kumar et al., “Prediction of Ayurvedic Herbs by Classification,” IEEE Conf., pp. 1204-1208, 2017.

L. Zhang et al., “Herb-Drug Interactions: Challenges,” Clinical Pharmacology, vol. 97, no. 6, pp. 560-570, 2015

TURNITIN PLAGIARISM REPORT



Unnati

Minor_Project_Final_Report Unnati.docx

Madhav Institute of Technology & Science

Document Details

Submission ID

trn:oid:::28506:121438253

Submission Date

Nov 14, 2025, 3:50 PM GMT+5:30

Download Date

Nov 14, 2025, 3:53 PM GMT+5:30

File Name

Minor_Project_Final_Report Unnati.docx

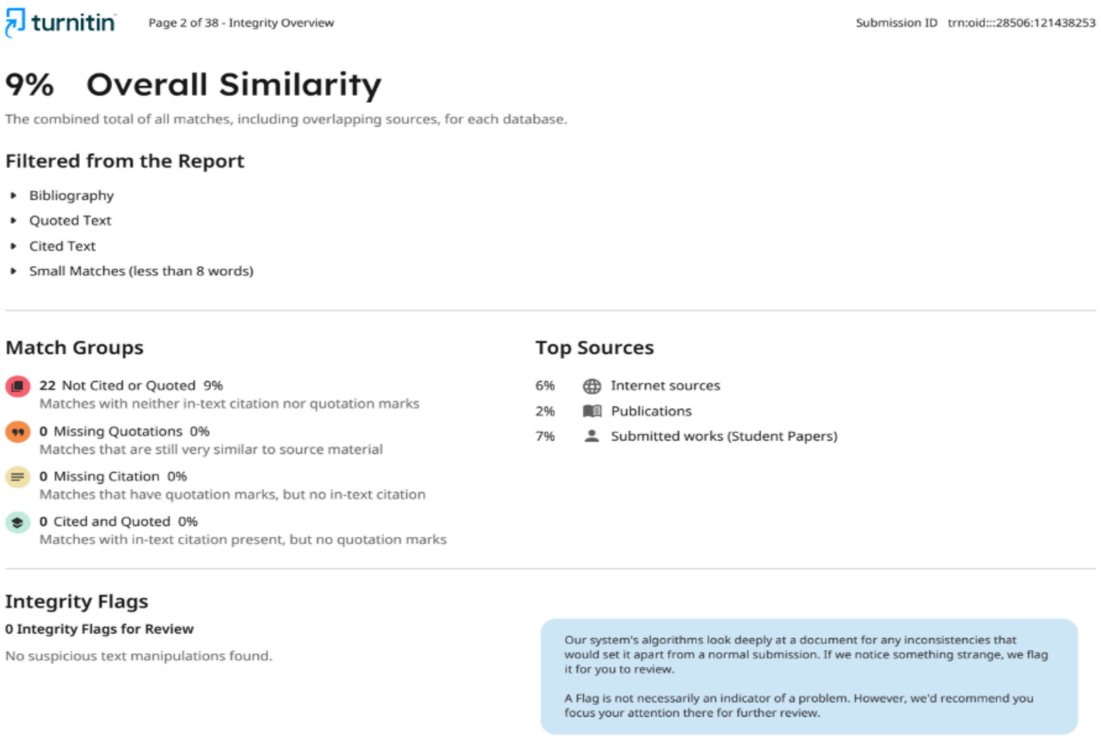
File Size

5.4 MB

34 Pages

4,625 Words

27,754 Characters



turnitin Page 2 of 38 - Integrity Overview Submission ID trn:oid:::28506:121438253

9% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

Filtered from the Report

- Bibliography
- Quoted Text
- Cited Text
- Small Matches (less than 8 words)

Match Groups

22 Not Cited or Quoted 9%
Matches with neither in-text citation nor quotation marks

0 Missing Quotations 0%
Matches that are still very similar to source material

0 Missing Citation 0%
Matches that have quotation marks, but no in-text citation

0 Cited and Quoted 0%
Matches with in-text citation present, but no quotation marks

Top Sources

6%	Internet sources
2%	Publications
7%	Submitted works (Student Papers)

Integrity Flags

0 Integrity Flags for Review

No suspicious text manipulations found.

Our system's algorithms look deeply at a document for any inconsistencies that would set it apart from a normal submission. If we notice something strange, we flag it for you to review.

A Flag is not necessarily an indicator of a problem. However, we'd recommend you focus your attention there for further review.