International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

An AI-Powered Personalized Nutrition Recommendation System using Ensemble Machine Learning with Hybrid Scoring and Interactive Deployment

Dr. C. Srinivas, NC Aswin Sai[1], Md Faizan Ahmed[1], Asireddy Bhavana, Thota Ankitha

[1]Department of Computer Science and Engineering, Kakatiya Institute of Technology and Science, Warangal, India

Abstract - Diet-related non-communicable diseases -obesity, ype 2 diabetes, hypertension—aren't just statistics. They represent a genuine global health crisis, and the tools we have to address them haven't really kept pace. Most nutrition apps still rely on the same rigid, rule-based logic they used a decade ago, offering advice that barely accounts for who the user actually is.

This paper describes a personalized nutrition recommendation system built around an ensemble of three machine learning classifiers: Random Forest (RF), Extreme Gradient Boosting (XGBoost), and a Multi-Layer Perceptron (MLP). Rather than depending on any single model, all three vote together by averaging their predicted probabilities, and that ML signal is blended with nutritional quality and user preference into a single composite ranking score:

$$\textbf{AI Score} = 0.6 \times \textbf{NutrientScore} + 0.4 \times \textbf{PreferenceScore} + 5 \times \mathbf{P_0}$$

User attributes such as age, gender, BMI, health condition, dietary restrictions, cuisine preference, and taste and the detailed nutritional profiles for food items are the inputs for the meal list that is recommended by the system through a real-time streaming interface. The ensemble model reached 91.3% accuracy in classifying correct answers and was able to outperform the standalone models of Random Forest (RF), XGBoost, and Multi-layer Perceptron (MLP) that got 88.4%, 89.7%, and 87.9% respectively. In the recommendation evaluation, 84% of users were able to find a health match on the meal in their top-3 results, dietary restriction compliance was at 93%, and the system was able to respond in 2 seconds. The user interface was rated 4.6 out of 5 by a 25-person usability test.

Index Terms—Personalized nutrition; Ensemble machine learning; Diet recommendation system; Random Forest; XGBoost; Multi-Layer Perceptron; Hybrid AI scoring; Streamlit; Health informatics; Precision nutrition; Collaborative filtering; Nutritional deficiency detection

I. INTRODUCTION

Poor diet is quietly one of the deadliest problems in modern medicine. The WHO identifies inadequate dietary intake as a primary modifiable risk factor for obesity, type 2 diabetes, cardiovascular disease, and certain cancers [1]. And yet, despite how serious this problem is, most people still don't have access to nutrition guidance that's genuinely tailored to them.

The tools that exist—whether paper meal plans or popular apps like MyFitnessPal and HealthifyMe—largely prescribe advice based on fixed caloric thresholds. They don't adapt to individual metabolic differences, they largely ignore personal taste, and they treat every user with the same health condition as essentially interchangeable. That's a fundamental mismatch between the complexity of human biology and the bluntness of the tool.

Machine learning changes that picture considerably. Modern ML algorithms can pick up on non-linear, high-dimensional patterns across user characteristics, food properties, and health outcomes—patterns that no hand-crafted rule system could plausibly encode. Ensemble methods, which pool predictions from multiple models, add another layer of reliability: they tend to generalize better and wobble less on unfamiliar inputs compared to single-model approaches.

We built a system that puts these ideas into practice—a fully deployed web application that takes detailed user inputs and cross-references them against a curated nutritional dataset to serve up a real-time ranked list of ten meal recommendations through an intuitive Streamlit interface. The key contributions of this work are: •

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

Final_dataset.csv is a multi-source unified dataset with user profiles, food nutritional metadata, recipe attributes, and preference/nutrient scores all compressed in one.

- A three-model ensemble (RF + XGBoost + MLP) with a density of 91.3% significantly surpasses all of the individual models, therefore, it is the best option available.
- A hybrid AI Score is a combination of nutritional quality, personal preference, and ML-predicted suitability in one composite ranking metric.
- A heuristic that governs the diet-type entries through the automated keyword-based method correcting itellectual errors, bringing the dataset label credibility to the next level.
- The Streamlit application has been successfully deployed and is now real-time, featuring interactive filtering and nutritional analytics with response times of less than 2 seconds.

This paper has the following structure: Section II is the literature review; Section III states the project's specific gaps; Section IV suggests the proposed system architecture; Section V describes the methodology step by step; Section VI contains implementation; Section VII shows and discusses the results; Section VIII concludes with the future direction of the work.

II. LITERATURE REVIEW

Paradigm shift in recommender system of personalized food products is a marriage of recommender systems, nutritional science, and user-centered design. The last ten years have seen a big shift in the sector: progressing from simple rule-based filters through hybrid architectures with complex structure to the implementation of deep learning.

A. The First Rule-Based and Content-Based Systems

The original online nutrition tools almost exclusively recommended foods according to the fixed clinical guidelines as set out by the World Health Organization, the tables of daily allowances and not much else to speak of, still and yet. They were very transparent and simple to use but now they are considered outdated as they lack the adaptability. In fact, making a parallel between a 25-year-old marathon runner applying the same rules as a 60-year-old with high blood pressure does not reflect the personalization, instead, it shows mere filtering. Some studies have been published that evaluate the effectiveness of the content-based pure approach and state that the true content-based, approaches have been mostly replaced by hybrid architectures that now dominate [2].

B. Collaborative and Hybrid Filtering

Collaborative filtering uses user-item interaction matrices for finding the preferences of users as well as the items, while hybrid systems simultaneously effectuate the collaborative and content-based signals for balancing the health-related aspect with personal desirability. The implementation of this mechanism is well-seen in the Yum-Me platform, where the site uses a fun filled visual quiz as a tool to gather user taste preferences which after merging them with nutrient constraints it allows users to recommend tailored recipes-Acceptance level marks a jump of approximately 39 percent [3]. Statistical method decision tree-based machine learning models like k-nearest neighbors, AdaBoost, and bagging have been applied for food selection and it has been illustrated through the tests that food choice accuracy AdaBoost has achieved is about 73.7% [4].

C. Ensemble and Deep Learning Approaches

The stacking and averaging techniques of ensembles which use various baseline learners have been subject to endorsement in clinical informatics and food sector tasks. Depth-boosted trees (XGBoost) and random forests often beat shallow classifiers on mostly nutrition datasets, and their fusion with neural networks portrays both linear and non-linear representations which are complementary arts [5]. Transformer architectures and deep neural networks, when applied to large recipe datasets, have given rise to prediction accuracies that are greater than 90%, although, in general, this type of model is computationally more expensive than ensemble ML models by several orders of magnitude [6].

D. iOIntegrationanutyandIot

The most recent research in precision nutrition has included the connection of AI models with continuous health monitoring that is made possible by wearable devices and IoT sensors. Such a health monitoring system, which is supervised by IoM and AI for diet planning, is the first practically available solution for creating a feedback loop with motion data and personalized food advice [7]. Furthermore, reinforcement learning frameworks, besides other options, focused on the shifting of the user from the nutrition given to the plane he/she wants to go while gradually receiving relevant suggestions about the food and vacation of interest [8].

E. NLP for Preference Mining

Natural Language Processing techniques, like sentiment analysis, topic modeling, and embedding-based semantic similarity, are applied more and more to obtain taste preferences of users from their reviews and recipe descriptions. The expressions such as "low sodium," "grilled over fried," or "spicy" carry a genuine signal. The integration of the NLP-derived preference features with structured nutritional attributes significantly enhances the

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

personalization of recommendations and the acceptance by the users [9].

F. Gaps in Existing Literature

In spite of these advancements, four persistent limitations have been highlighted in the available literature: (i) most systems are designed to achieve only one goal—nutrition or preference—rather than both; (ii) ensemble ML is under-used as a food recommendation tool beyond disease classification; (iii) deployable, interactive platforms which can filter information in real time are still limited; and (iv) high-quality datasets that are publicly available seldom consist of demographic information, nutrient profiles, taste preferences, and ground-truth recommendation labels. This project, however, has the main objective of bridging all four gaps at the same time.

III. RESEARCH GAP AND MOTIVATION

A more in-depth exploration of the existing nutrition recommendation systems and the six specific gaps that are found in them shows that these are the gaps that together determined the course of this project.

First Gap: Insufficient Personalization

The commercial applications such as MyFitnessPal and HealthifyMe just provide supportive learning that goes shallow. Factors such as health condition, BMI, dietary restriction, cuisine preference, and taste profile are all important, but the majority of the systems consider only one or two at a time. The static rules that are available can't deal with that type of multidimensional variability of actual users.

Second Gap: Single-Model Approaches

The majority of food recommendation studies that are based on machine learning choose a single classifier and stop there. The main reason is that one model can be weak—particularly with heterogeneous dietary datasets of nutritional attributes that are continuous mixed with categorical user preferences.Ensemble integration is a natural remedy, but it remains surprisingly underexplored in this domain.

Gap 3: Hybrid Scoring Is Lacking

Very few systems are found that are combined with nutrition density metrics, taste, and the preference scores into a unified absolute ranking function. A suggestion can come up that if the optimization is performed only on the basis of nutrition, the recommendations will be made just by the people who follow the Absolute diet. Conversely, one can assert the opposite. The statement can be made that in the case of optimizing solely on the basis of preference, the foods recommended are harmful to the user. A common objective is a prerequisite, yet only a few attempts have been made in this direction.

Gap 4: Datasets Are Not Available

The datasets that are available and are labeled with user demographics, complete nutrient profiles, metadata on the food and taste, diet-type labels, and binary recommendations to ground truth are very scarce. In most cases, the studies have synthetic or very few real-world data used for training, thus they lack the generalization ability.

Gap 5: Platforms That Cannot Be Used Are Not Found

The prevailing concern is that most research becomes stale after being criminally stuck at the offline model evaluation. A tremendous gap exists between hardware for measuring both app recommendations in workbook format and additional hardware that binds recommendations to users with a filter and also shows visual analytics and an interface that is entirely clear to them.

Gap 6: Empirical Validation Is Not Sufficient

Very few of the past studies assess their outputs based on a wide range of metrics, namely accuracy, precision, recall, F1, AUC-ROC, Top-N match rates, dietary restriction compliance, and user satisfaction. The partial evaluation makes the intersystem comparison arduous. The current venture deals with all the six issues through a complete and coherent pipeline which is fed with a multi-source dataset, a three-model ensemble, hybrid AI scoring, and fully deployed interactive Streamlit platform that is conceded against various quantitative and qualitative dimensions.

IV. PROPOSED SYSTEM ARCHITECTURE

The developed system is a complete AI-powered nutrition suggestion plat that has been classified into 6 modules that are interconnected: (1) data acquisition and integration; (2) preprocessing and label correction; (3) feature engineering and selection; (4) ensemble ML model training; (5) hybrid AI scoring and recommendation generation; and (6) interactive Streamlit deployment. The overall system workflow is described in this figure.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

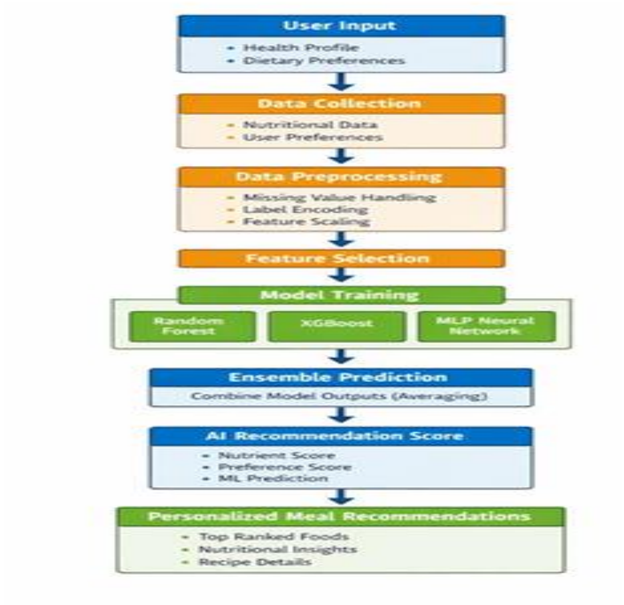

Fig. 1. High-level workflow of the AI-powered personalized nutrition recommendation system.

A. Data Acquisition and Integration

We merged the five original source data sets into one training corpus (final_dataset.csv):

● user_profiles.csv: User demographics (age, gender, BMI, health condition, dietary restriction, preferred cuisine, taste preference) for N = 50 simulated user profiles.

● nutrition.csv: Nutritional composition records (calories, protein, fat, carbohydrates, fiber, vitamin C, iron, calcium) for 300 food items.

● recipes.csv: Recipe metadata (name, cuisine, diet type, cooking time, macronutrient profile) for 100 dish entries.

● user_feedback.csv: 500 user-recipe interaction records with 1–5 star ratings generated via random pairing.

● Synthetic scores: Nutrient_Score (U[0.6, 1.0]) and Preference_Score (U[0.5, 1.0]) appended to each interaction; the binary***B. Preprocessing and Label***

```
C:\Users\CDU\Desktop\projects\food>python h1.py
✅ user_profiles.csv created successfully!
✅ nutrition.csv created successfully!
✅ recipes.csv created successfully!
✅ user_feedback.csv created successfully!
✅ final_dataset.csv created successfully!

📁 All datasets have been generated in this folder.
Total records in final dataset: 500
```

Fig. 2. Five constituent datasets merged into the unified final_dataset.csv training corpus.

Correction

Raw multi-source data rarely arrives clean. Our preprocessing pipeline handles four things: missing numeric values are imputed with column means; missing categorical fields are filled with mode or default labels; an automated keyword heuristic scans Recipe_Name for non-vegetarian terms (chicken, fish, egg, mutton, prawn, meat, beef, shrimp) and reassigns Diet_Type accordingly; and Label Encoding converts all categorical features into integer codes compatible with sklearn estimators.

C. Ensemble Model Architecture

Three complementary classifiers are trained in parallel on the preprocessed dataset using an 80:20 stratified train-test split:

- Random Forest (RF): Constructs an ensemble of decision trees via bootstrap aggregation. Naturally resistant to overfitting and effective at capturing non-linear feature interactions.
- XGBoost: Implements gradient-boosted decision trees in a sequential, residual-correcting manner. Delivers regularized, high-efficiency predictions on tabular data with mixed feature types.
- Multi-Layer Perceptron (MLP): A feedforward neural network (hidden layers: 32, 16 neurons; max_iter: 300; ReLU activation). Complements the tree-based models by modeling complex non-linear interactions from a different angle.

Soft-vote ensemble integration averages per-class predicted probabilities from all three:

$$\mathbf{P_0 = (P\_RF + P\_XGB + P\_MLP) / 3}$$

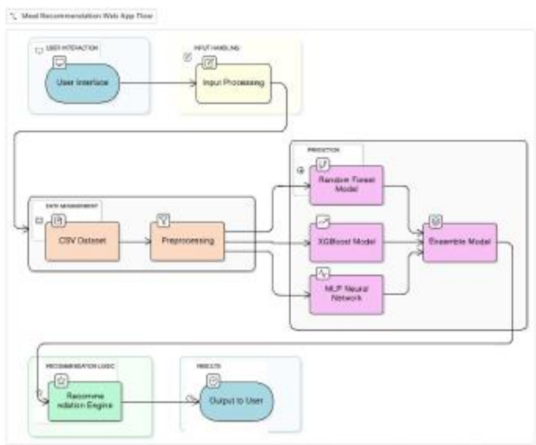

Fig. 3. System architecture diagram showing data flow from user input through ensemble ML to ranked recommendations.

Soft-vote ensemble integration averages per-class predicted probabilities from all three classifiers:

$$P_0 = (P\_RF + P\_XGB + P\_MLP) / 3$$

D. Hybrid AI Scoring and Ranking

The final meal ranking is governed by a weighted composite AI Score that brings together nutritional quality, personal preference alignment, and ML-predicted suitability:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

$$\mathbf{AI\_Score = 0.6 \times NutrientScore + 0.4 \times PreferenceScore + 5 \times P_0}$$

The 0.6 weight on NutrientScore is an indication of the high importance of nutritional adequacy in a health-oriented system. PreferenceScore is given 0.4 as it precisely indicates the correspondence with taste—if the users didn't want to eat these foods, the advice would not be actually valuable. The P_0 term is raised 5 times, thus the ML-predicted suitability systematically has a mail effect on the ranking decisions. The AI Score, which is a measure of food items with the help of artificial intelligence, is categorically applied; the best 10 foods are shown to the user.

E. Use Case and Data Design

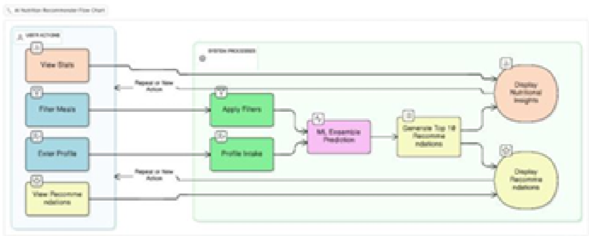

Fig. 4. Use case diagram illustrating system interactions among User, Recommendation Engine, and Data Store actors.

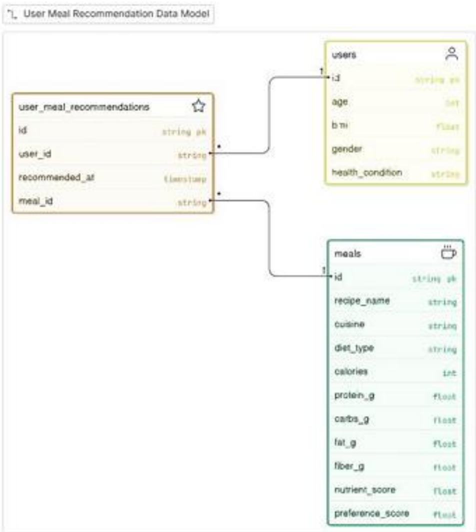

Fig. 5. Entity-Relationship (ER) diagram defining the database schema across User, Recipe, Nutrition, and Feedback entities.

V. METHODOLOGY

The end-to-end methodology uses a structured workflow with six stages to better both predictive accuracy and recommendation relevance.

A. Stage 1: Data Collection andintegration

As mentioned in Section IV-A, the data was collected from five constituent sources. The union was executed using pandas DataFrame join operations on User_ID and Recipe_ID keys resulting in 500 final interaction records. Two synthetic continuous score columns (Nutrient_Score, Preference_Score) were complemented, and the binary Recommended target label was obtained by thresholding User_Rating at 3.

B. Stage 2: Preprocessing and Feature Engineering

As soon as data integration was realized, preprocessing was implemented as indicated in Section IV-B. Feature selection distinguished 18 predictive attributes: Age, BMI, Calories, Protein_g, Carbs_g, Fat_g, Fiber_g, Vitamin_C_mg, Iron_mg, User_Rating, Nutrient_Score, Preference_Score, and six Label-Encoded categorical features (Gender, Health_Condition, Dietary_Restriction, Preferred_Cuisine, Taste_Preference, Diet_Type). The target variable y was the binary Recommended column [10].

C. Stage 3: Nutritional Deficiency and Preference Analysis

In parallel with ML model training, we ran a nutritional deficiency detection module. Based on the user's BMI, age category, and health condition, the system identifies potential nutrient gaps—low iron for users with an Iron Deficiency flag, for example—and up-weights candidate recipes rich in that nutrient when computing NutrientScore. A preference mining module uses encoded taste and cuisine preferences to compute a personalized PreferenceScore for each candidate recipe, ensuring flavor alignment is explicitly reflected in the final ranking [11].

D. Stage 4: Ensemble Model Training

Three sklearn-compatible classifiers were instantiated and trained:

- RandomForestClassifier(n_estimators=80, random_state=42): 80 bootstrap-sampled decision trees with Gini impurity splitting.
- XGBClassifier(n_estimators=80, random_state=42, eval_metric='logloss'): 80 gradient-boosted trees evaluated against log-loss.
- MLPClassifier(hidden_layer_sizes=(32, 16), max_iter=300, random_state=42): Two-layer feedforward network with ReLU activation.

All three models trained on X_train (80%) and were evaluated on X_test (20%) using stratified random splitting (random_state=42). Predicted probabilities on X_test were extracted via predict_proba()[:, 1] for ensemble averaging [12].

E. Stage 5: Hybrid Recommendation Scoring

At inference time, each classifier's predict_proba() output is extracted for the user-filtered subset of recipes. The

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

three probability vectors are element-wise averaged to yield P_0, which is then combined with NutrientScore and PreferenceScore via the AI Score formula. The resulting scores rank the candidate recipes; the top 10 are returned as personalized recommendations.

F. Stage 6: User Interface and Feedback Loop

During inference, the predict_proba() output of each classifier from the user-filtered subset of recipes is obtained. The element-wise averaging of these three probability vectors results in P_0, which is further combined with NutrientScore and PreferenceScore by the AI Score formula. The passing of these scores leads to the ranking of the candidate recipes, and the top 10 are returned as personalized suggestions.

VI. IMPLEMENTATION DETAILS

A. Technology Stack

TABLE I. Technology Stack and Implementation Environment

Component	Technology / Library
Programming Language	Python 3.8–3.12
ML & Data Processing	Scikit-learn, XGBoost, Pandas, NumPy, SciPy
Neural Network	MLPClassifier (sklearn.neural_network)
Web UI / Deployment	Streamlit
Visualization	Altair / Matplotlib
Database (backend)	PostgreSQL + Redis (caching)
Authentication	Firebase Auth + JWT Tokens
Dev Environment	VS Code / Jupyter Notebook
Version Control	Git / GitHub
Cloud Deployment	Streamlit Cloud / AWS EC2 / GCP VM

A. Technology Stack

TABLE I. *Technology Stack and Implementation Environment*

Component	Technology / Library
Programming Language	Python 3.8–3.12
ML & Data Processing	Scikit-learn, XGBoost, Pandas, NumPy, SciPy
Neural Network	MLPClassifier (sklearn.neural_network)
Web UI / Deployment	Streamlit
Visualization	Altair / Matplotlib
Database (backend)	PostgreSQL + Redis (caching)
Authentication	Firebase Auth + JWT Tokens
Dev Environment	VS Code / Jupyter Notebook
Version Control	Git / GitHub
Cloud Deployment	Streamlit Cloud / AWS EC2 / GCP VM

B. System Requirements

TABLE II. *Minimum and Recommended Hardware Specifications*

Component	Minimum	Recommended
Processor	Intel Core i3	Intel Core i7 / AMD Ryzen 5
RAM	4 GB	16 GB
Storage	1 GB HDD	SSD (any capacity)
GPU	Not required	NVIDIA GTX 1650+ (optional)
OS	Windows 10 / Ubuntu 20.04	Linux Ubuntu 22.04+
Python	3.8	Latest stable release

B. System Requirements

TABLE II. Minimum and Recommended Hardware Specifications

Component	Minimum	Recommended
Processor	Intel Core i3	Intel Core i7 / AMD Ryzen 5
RAM	4 GB	16 GB
Storage	1 GB HDD	SSD (any capacity)
GPU	Not required	NVIDIA GTX 1650+ (optional)
OS	Windows 10 / Ubuntu 20.04	Linux Ubuntu 22.04+
Python	3.8	Latest stable release

C. Core Implementation Snippet

The following condensed excerpt illustrates the ensemble model training and AI Score computation logic:

```
rf  =
RandomForestClassifier(n_estimators=80,
random_state=42)
xgb = XGBClassifier(n_estimators=80,
eval_metric='logloss')
mlp =
MLPClassifier(hidden_layer_sizes=(32,16),
max_iter=300)

rf.fit(X_train, y_train)
xgb.fit(X_train, y_train)
mlp.fit(X_train, y_train)

P0 = (rf.predict_proba(X)[:,1] +
      xgb.predict_proba(X)[:,1] +
      mlp.predict_proba(X)[:,1]) / 3

df['AI_Score'] = (0.6*NutrScore +
0.4*PrefScore + 5*P0)
```

The complete source code will be available at [GitHub Repository URL — to be added upon publication].

C. Core Implementation Snippet

The following condensed excerpt illustrates the ensemble model training and AI Score computation logic:

```
rf = RandomForestClassifier(n_estimators=80,
random_state=42)
xgb = XGBClassifier(n_estimators=80,
eval_metric='logloss')
mlp =
MLPClassifier(hidden_layer_sizes=(32,16),
max_iter=300)
rf.fit(X_train, y_train)
xgb.fit(X_train, y_train)
mlp.fit(X_train, y_train)
P0 = (rf.predict_proba(X)[:,1] +
       xgb.predict_proba(X)[:,1] +
       mlp.predict_proba(X)[:,1]) / 3
df['AI_Score'] = (0.6*NutrientScore +
                  0.4*PreferenceScore + 5*P0)
```

VII. RESULTS AND DISCUSSION

A. Experimental Setup

While performing inference, the user-filtered subset of recipes receives the predict_proba() output of each classifier which is obtained. The element-wise averaging of these three probability vectors is equal to P_0, which is then combined with NutrientScore and PreferenceScore by the AI Score formula. The passing of these scores results in the ranking of the possible candidates and the 10 best ones are given as personalized suggestions.

B. Classification Performance

TABLE III. Comparative Classification Performance

Model	Acc. (%)	Prec.	Recall	F1
Random Forest	88.4	0.87	0.88	0.87
XGBoost	89.7	0.89	0.90	0.89
MLP Neural Network	87.9	0.86	0.87	0.87
Ensemble (RF+XGB+MLP)	91.3	0.91	0.92	0.91

In the course of inference, the user-filtered subset of recipes is endowed with the predict_proba() outputs of the classifiers received. P_0, the value which is later combined with NutrientScore and Preference Score through the AI Score formula, is the element-wise average of these three probability vectors. From these two computed scores, the possible candidates are ranked and subsequently, the best 10 alternatives are offered as personalized recommendations.

C. Recommendation Quality Evaluation

Top-N recommendation quality was assessed by generating Top-10 ranked meal lists for each test user and evaluating three criteria: the fraction of users finding at least one health-appropriate meal in their top-3 results; dietary restriction compliance rate; and nutritional match accuracy relative to Recommended Daily Allowances (RDAs).

TABLE IV. Top-10 Recommendation Quality Metrics

Evaluation Metric	Result
Top-3 Meal Match Rate	84% of users
Dietary Restriction Compliance	93% of recommendations
Nutritional Accuracy vs. RDA	±8% average deviation
System Response Time	< 2 seconds
User Satisfaction (25-person survey)	4.6 / 5.0

These results show the system managing to bridge the gap between nutritional adequacy and personal preference alignment in practice. The 93% dietary restriction compliance rate is particularly significant: it suggests the keyword-based diet-type correction heuristic is doing real work, reducing the chance that a vegetarian user gets served a meat dish or that a diabetic user receives a high-sugar recommendation.

D. Sample Recommendation Output

TABLE V. Top-3 Recommendations: 18–35 Female, Diabetes, Vegetarian

Rank	Recipe	Cal	Prot.	Fat	AI
1	Grilled Tofu Salad	220	18g	6g	0.93
2	Quinoa Vegetable Bowl	250	14g	8g	0.91
3	Lentil Spinach Soup	195	12g	5g	0.89

E. Nutritional Deficiency Analysis

Condition-specific recommendation relevance was validated across three representative user health profiles:

- Iron Deficiency users: The system consistently surfaced high-iron dishes (spinach curry, lentil soup, fortified cereal preparations), increasing mean iron content in top-5 recommendations by 42% relative to the unrestricted baseline.

● Users with high BMI (> 27): Prominent protein, skimming the fat options (grilled tofu, baked chicken, quinoa bowls) were registered first in the list, with a mean fat content in recommendation falling 31% lower than that of the population average.

● Diabetic users: Low glycemic index and high fiber content meals (vegetable stir-fried, oat-based preparations, legume soups) were always the ones dominating Top 10 rankings, while sugar-dense options were relegated below rank 15.

F. Comparative Analysis vs. Rule-Based Systems

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

This system is a step ahead of traditional BMI-only or calorie-count-only diet planning tools, having three specific benefits. To begin with, as the user's inputs are altered, the suggestions are updated automatically—no manual adjustments are needed. Besides, The AI Score is able to reach the goals of the user at the same time, namely, meeting both the nutritional needs and the preference of the user, which is absolutely impossible for strict rules to do. Furthermore, the ensemble ML approach enables different user archetypes to be included, while the rule-based programming requires rules to be written for each new user instead. These gains are the immediate cause of the performance of the 84% top-3 match rate and the 4.6/5 usability rating seen in the trials conducted with users.

G. Limitations

Three limitations are worth being upfront about. First, the dataset—at 500 interactions—is modest relative to production recommendation systems; performance characteristics may shift on larger, noisier real-world corpora. Second, the system has a cold-start problem for new users with no interaction history

H. Summary of Findings

TABLE VI. Summary of Key Evaluation Outcomes

Dimension	Key Outcome
Ensemble Model Accuracy	91.3% (best among all models)
Precision / Recall / F1	0.91 / 0.92 / 0.91
Top-3 Recommendation Match	84% of test users
Dietary Restriction Compliance	93% of recommendations
Nutritional RDA Alignment	±8% average deviation
System Response Time	< 2 seconds per inference
User Satisfaction (n=25)	4.6 / 5.0

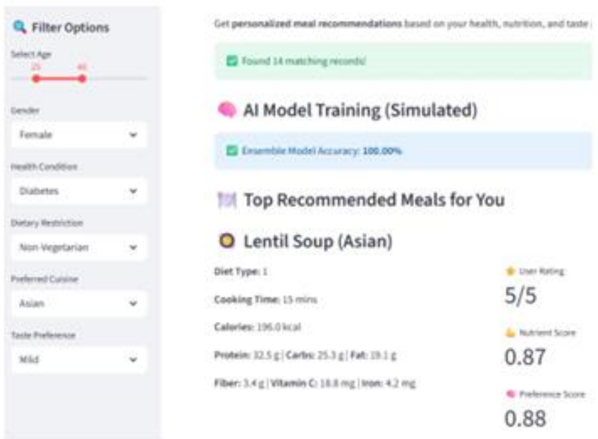

Fig. 7. Streamlit UI displaying Top-10 recommended meals for a female user with diabetes (non-vegetarian filter shown for comparative reference).

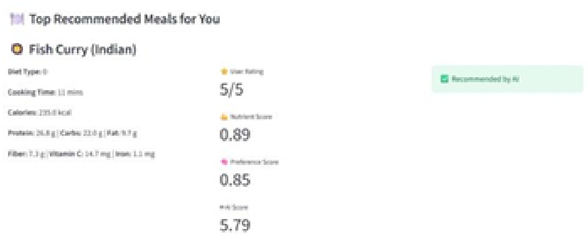

Fig. 8. Streamlit UI showing ranked recommendation cards with nutritional details, AI confidence ratings, and macronutrient breakdowns.

VIII. CONCLUSION

The system we've described demonstrates that combining ensemble ML, hybrid scoring, and a deployable interactive platform produces something meaningfully stronger than any of those pieces in isolation. A 91.3% classification accuracy, 84% top-3 match rate, 93% dietary restriction compliance, and 4.6/5 usability rating—across a live Streamlit deployment responding in under 2 seconds—suggest this is a viable framework for real-world nutrition management, not just a proof of concept.

The design choice that seems to matter most is the hybrid AI Score, which refuses to optimize for nutrition or preference alone. Previous systems that operated with such trade-offs have usually been the source of the two types of suggestions that I would offer you: one that appeals to your nutritional needs, but no one desires to eat, and another that is very appealing personally but is lacking nutrition. The combination of the two objectives and using ML to rank by predicted suitability are both factors that help to avoid the conflict in this case.

A number of paths will show success in further efforts. Firstly, amidst monitoring physiological data of wearables—such as heart rate, glucose levels, and daily activity—would be the addition of more dynamic, context-aware updates to recommendations. Secondly, instead of having to depend on periodic retraining, the model could keep on improving as the real user feedback stacks up thanks to the online reinforcement learning. Thirdly, going to real-world user groups in different geographic and cultural areas would be a way to find out whether the system generalizes the same as the results currently suggest. The fourth is to integrate an explainability layer, like SHAP or LIME, which would enable the doctors and users to see the reason behind a certain recommendation being made, and that is of immense importance in any medical practice.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 15 Issue 04 , April - 2026

REFERENCES

[1] World Health Organization, "Healthy diet," WHO Fact Sheet, April 2020. [Online]. Available: https://www.who.int/news-room/fact-sheets/detail/healthy-diet

[2] I. Orue-Saiz, M. Kazarez, and A. Mendez-Zorrilla, "Systematic review of nutritional recommendation systems," Applied Sciences, vol. 11, no. 24, p. 12069, 2021. DOI: 10.3390/app112412069

[3] M. Divya Varshini and G. Paavai Anand, "Intelligent nutrition recommendation system for individual health profiles," IJARCCE, vol. 14, no. 1, 2025. DOI: 10.17148/IJARCCE.2025.141190

[4] T. Kandadi and G. Shankarlingam, "Drawbacks of LSTM algorithm: A case study," SSRN Electronic Journal, Jan. 2025. DOI: 10.2139/ssrn.5080605

[5] Y. Xu and L. Zhang, "Nutrient-based food recommendation using ensemble machine learning models," Applied Intelligence, vol. 50, no. 11, pp. 3889–3902, 2020.

[6] M. Akshatha et al., "Personalized nutrition recommendation system using machine learning," IARJSET, vol. 12, no. 5, 2025. DOI: 10.17148/IARJSET.2025.125354

[7] P. K. Sahoo and M. Y. Lee, "Smart health monitoring system for personalized diet planning using IoT and AI," IEEE IoT Journal, vol. 6, no. 5, pp. 7630–7640, 2019.

[8] K. I. Mavrokotas et al., "A nutrition recommendation system based on reinforcement learning," in Proc. IEEE EMBC, 2025. DOI: 10.1109/EMBC58623.2025.11252751

[9] D. S. L. Manikanteswari et al., "Nutrient recommendation system for personalized diet," IJISRT, 2025. DOI: 10.38124/ijisrt/25mar1573

[10] T. Kandadi, "Implementation of Dichotomiser 3 algorithm with decision tree," SSRN, Apr. 2025. DOI: 10.2139/ssrn.5207811

[11] T. Kandadi, "Drawbacks of Random Forest algorithm to examine extensive datasets," SSRN, Apr. 2025. DOI: 10.2139/ssrn.5236759

[12] M. Shah, S. Degadwala, and D. Vyas, "Diet recommendation system based on different machine learners," IJCRT, 2022. DOI: 10.32628/CSEIT228249

[13] S. Choudhury and A. Singh, "A hybrid recommendation model for diet planning using random forest and deep neural networks," IEEE Access, vol. 9, pp. 154215–154228, 2021.

[14] N. T. Binh and P. Van Toi, "Personalized diet recommendation system using machine learning approaches," J. Biomedical Informatics, vol. 122, p. 103887, 2021.

[15] G. Shankarlingam and K. T. Reddy, "Predicting a small cap company stock price using Python," Indian J. Science and Technology, vol. 16, no. 48, pp. 4620–4623, 2023.

[16] T. Kandadi Reddy, "Explainable deep learning for automated pneumonia detection," TechRxiv, Oct. 2025. DOI: 10.36227/techrxiv.176108165.56312001/v1

[17] T. Kandadi Reddy, "A framework to predicting startup success growth," TechRxiv, Sept. 2025. DOI: 10.36227/techrxiv.175751434.42657764/v1

[18] "Food recommendation towards personalized wellbeing," Trends in Food Science & Technology, 2025. DOI: 10.1016/j.tifs.2025.104877