

Proceeding Paper

Multi-Choice Diet Recommendation Application for Indian Scenario Based on Insights from Ensemble Learning Techniques [†]

Karthika Subbaraj 

Department of Information Technology, Sri Sivasubramaniya Nadar College of Engineering, Kalavakkam, Chennai 603110, India; skarthika@ssn.edu.in

[†] Presented at the 2nd Computing Congress 2023, Chennai, India, 28–29 December 2023.

Abstract: Indian diet has changed over the years as a result of geographic, cultural, and traditional influences. However, Western influences have recently had a substantial impact, including changes in eating behaviors and an increase in the consumption of fast food. Adoption of these foods has also led to a shift from traditional Indian meals, which mostly consists of a variety of grains, vegetables, lentils, and spices. In contrast to fast food, which is frequently highly processed and deficient in important nutrients, these traditional diets are frequently more wholesome, balanced, and nutritious. Obesity rates in India have risen rapidly, with a significant increase in cases of diabetes, heart disease, and other diet-related illnesses. According to the World Health Organization (WHO), the occurrence of obesity tripled between 1975 and 2016 and is continually rising. This research work seeks to give customers a meal plan that complies with their calorie needs based on their BMI and TDEE. The meal plan will only include foods that are available in India. A dish may also be substituted if the user does not have access to it. The meal plan is generated using the random forest classifier and the alternatives are provided by using the KNN algorithm. The classifier has an accuracy of 91%.

Keywords: meal classification; machine learning; BMI; total daily energy expenditure; predictions



Citation: Subbaraj, K. Multi-Choice Diet Recommendation Application for Indian Scenario Based on Insights from Ensemble Learning Techniques. *Eng. Proc.* **2024**, *62*, 13. <https://doi.org/10.3390/engproc2024062013>

Academic Editors: Geetha Ganesan, Xiaochun Cheng and Valentina Emilia

Published: 18 March 2024



Copyright: © 2024 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Indian diet has evolved over centuries, influenced by culture, geography, and traditions. However, in recent times, it has been significantly impacted by Western influences, leading to changes in eating habits and the rise of fast food consumption [1]. Fast food, which is often high in unhealthy fats, sugars, and calories, has become increasingly popular in India due to factors like urbanization, globalization, and changing lifestyles. The World Health Organization (WHO) highlights that during the period between 1975 and 2016, the global occurrence of obesity has tripled. Obesity prevalence, formerly considered to be a problem in wealthy countries, is now changing in developing countries. The main causes of overweight and obesity among teenagers are inappropriate diet or lack of physical activity as stated by National Institutes of Health [2].

According to a report by the World Health Organization, being obese during childhood raises the chances of developing adult obesity and other non-communicable conditions such as cardiovascular disease, type II diabetes, musculoskeletal disorders, and specific cancers, including breast, liver, gallbladder, kidney, colon, endometrial, ovarian, and prostate cancers [2]. The consequences of weight issues in India are significant, including increased healthcare costs and reduced quality of life. In the World Obesity Atlas 2023 report, the data indicate that in 2020, the obesity risk for boys in India was 3 percent. However, it is projected to increase significantly to 12 percent by 2035. Similarly, for girls, the risk stood at 2 percent in 2020, but it is expected to climb to 7 percent in the next 12 years. Therefore, a change in eating habits is required and the application discussed aims to achieve this. India has become one of the concerned countries in this list. According to estimates, India has 135 million obese people.

This work proposes a meal plan for users according to their calorie requirements that is accessible to the public and user-friendly. The dishes must be classified according to the time they are eaten, and the meal plan should be generated accordingly. The objectives of this paper are, firstly, to identify a model such as random forest that classifies if a given dish is a breakfast, lunch, or dinner dish according to its macronutrient values and food group, and secondly to create a model that provides alternatives to a suggested item if a user requests the same.

2. Related Work

Garcia et al. conducted a study on different recommender systems in the healthcare domain [3]. The paper first started with a study on recommendation systems and their different aspects and techniques. The different techniques studied were collaborative filtering, content-based filtering, knowledge-based recommendation, and hybrid recommendation. Anaka Aiyar et al. designed a diet recommendation framework to overcome PCOS using deep convolutional neural networks [2]. The framework first uses a CNN algorithm to identify a picture of seven of the user's meals. To generate a meal plan, it uses the K-Means Algorithm to cluster their dataset. With the help of random forest algorithms, the authors identified the closest cluster according to nutritional requirements. The SVM, NB, and random forest models achieved notable levels of precision and accuracy. In a survey conducted by Absalom E. Ezugwu et al., various clustering algorithms were examined [1]. The paper covered a wide range of clustering algorithms, including hierarchical clustering methods and partitional clustering methods. Hierarchical clustering algorithms divide the items into classes, whereas partitional methods group items which are similar. Yanli Liu et al. described what random forests are in [4]. MF Raynor et al. developed a mobile-based personalized diet recommendation system using a random forest classifier model to predict daily calorie intake and macronutrient distribution [5]. Random forests are a combination of tree classifiers, where each tree casts a unit vote for the most popular class and combining the results provides the final sort result. Mifflin MD et al. provided an alternative and more valid estimate of BMR (Basal Metabolic Rate) [6,7]. The Harris–Benedict Equations, developed in 1919, were found to overestimate the measured Resting Energy Expenditure (REE) by 5%. This conclusion was drawn after a study involving individuals, during which REE was measured using indirect calorimetry. The system allows users to set dietary goals, track food intake, and receive personalized recommendations based on preferences and nutritional needs. Testing with 30 users showed the system effectively improved dietary habits and achieved weight loss goals, highlighting the potential of using machine learning for personalized dietary interventions through mobile applications. Nguyen PH et al. carried out collaborative filtering using the k-nearest-neighbor method [8]. The concept was to locate k neighbors in reverse. As fewer rating forecasts were made, discovering the nearby neighbors takes less time. Osisanwo F.Y et al. performed a comparison between the different classification algorithms [9]. The paper studied a total of nine algorithms, including Naive Bayesian (NB) Networks, Support Vector Machines, and Decision Trees to name a few. It was inferred that different algorithms are better for different datasets and it was difficult to find a single classifier that performed as well as an ensemble model.

3. Materials and Methods

This research work adapts the idea from [10] and provides the user with a meal plan for the day that adheres to their own individual calorie requirements. As this application is for Indians, the data needs to contain food that will be available in India and is easily accessible. The dataset of interest will be one that contains the nutritional details of food. However, there is a lack of comprehensive and reliable data on the nutrient content of Indian foods and there is a lack of standardization in the methods used to measure nutrient content, which makes it difficult to compare data across studies and to develop accurate nutritional databases for Indian foods. The proposed architecture of the recommendation system is as shown below in Figure 1.

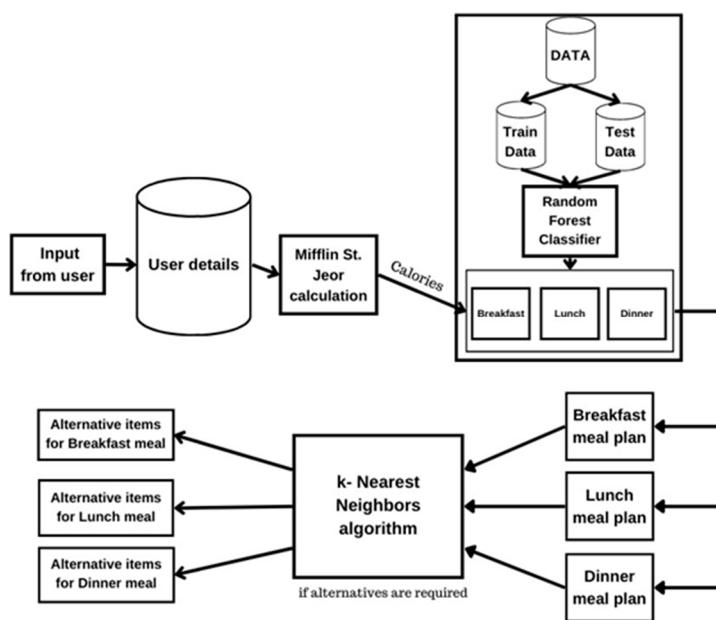


Figure 1. Overview of diet recommendation system.

The database used in this research was ‘MyFoodData’. It provides nutrition data which are sourced from the USDA Food Central created based on guidelines from [11]. The dataset contains 13,628 records with 118 features. All 118 features were not used for the classification model that was to be trained. To understand how closely the features were correlated to each other, a heatmap was generated using the seaborn library as shown in Figure 2.

The classification of all the dishes and the curated meals to be generated are based on numerical values. Thus, to provide results for the Indian diet, the classification model, i.e., the random forest classifier, was trained on the records where the value of the feature ‘Indian’ was False. This model was then used to predict the meals for the records where the value of the feature ‘Indian’ was True.

The train test split was performed manually by making two different data frames, ‘train_data’ and ‘test_data’, where the former has all records where the value under the feature ‘Indian’ is False and the latter has all records with a True value. It was inferred that fat, proteins, and carbohydrates were closely related to calories. To understand what features to use for classifying a dish into a meal type, pair plots were visualized. [0,0,0] exists, in the following table, because the dataset procured contained many unhealthy food items, which goes against the main objective of the application being created.

In the dataset, there were 4 meal types, as shown in Table 1.

Table 1. Meal type.

String Used in Dataset	Meal
1,0,0	Breakfast
0,1,0	Lunch
0,0,1	Dinner
0,0,0	Unhealthy

The pair plots against the features ‘Protein (g)’, ‘Fat (g)’, ‘Carbohydrate (g)’, and ‘Food Group’ grouped by ‘Meal’ are given below in Figures 3 and 4.

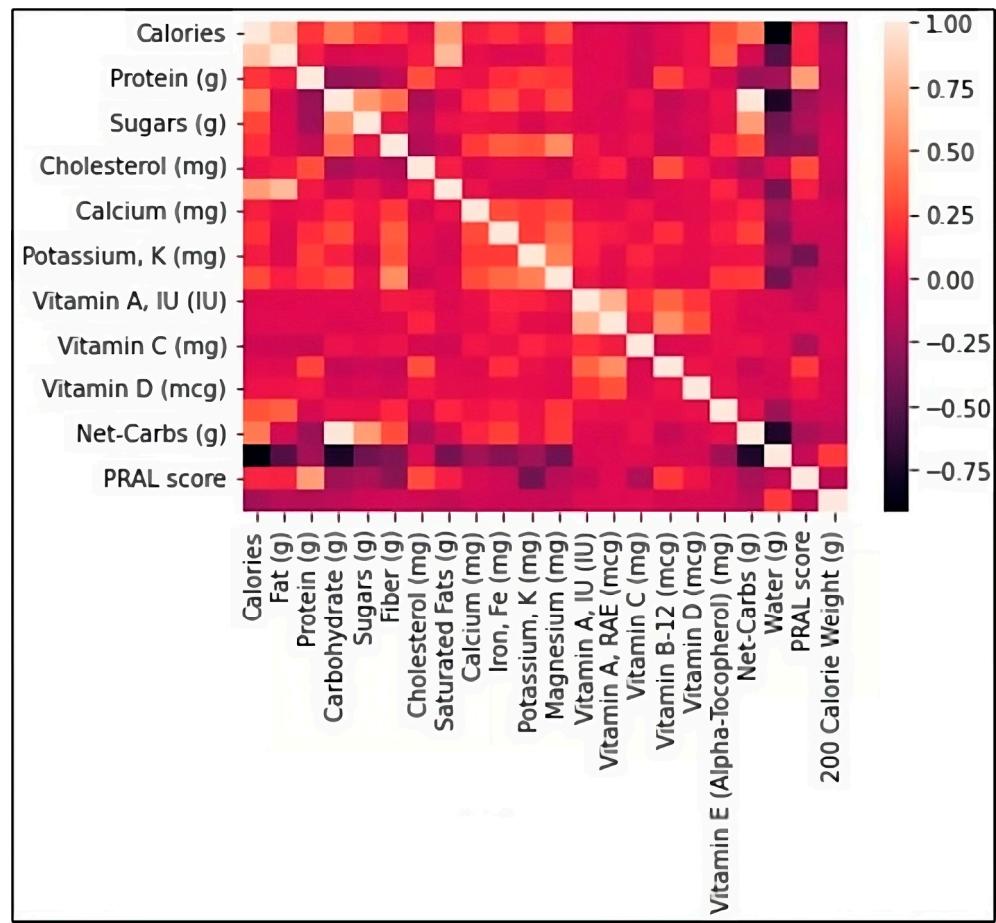


Figure 2. Feature correlation map.

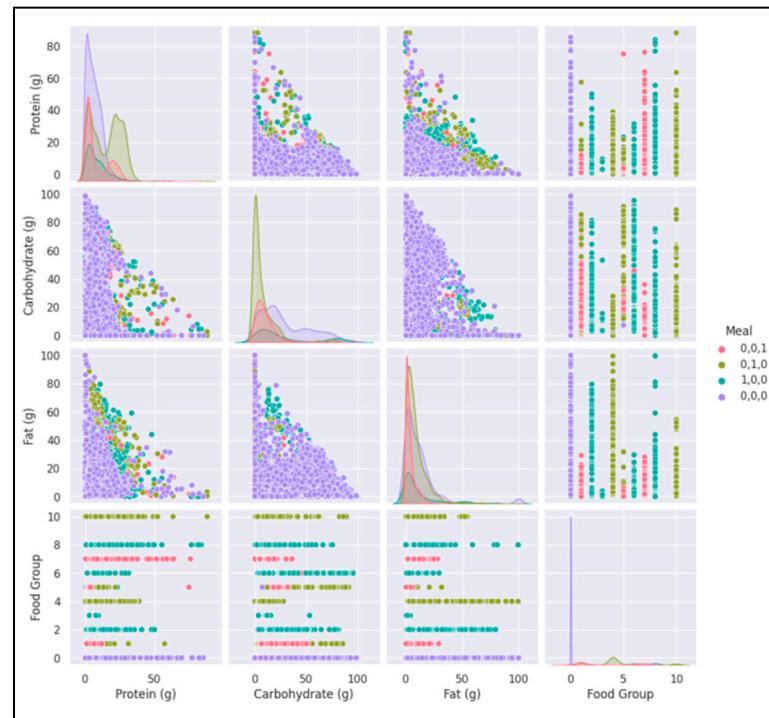


Figure 3. Pair plot for macronutrients and meal with 0,0,0 value.

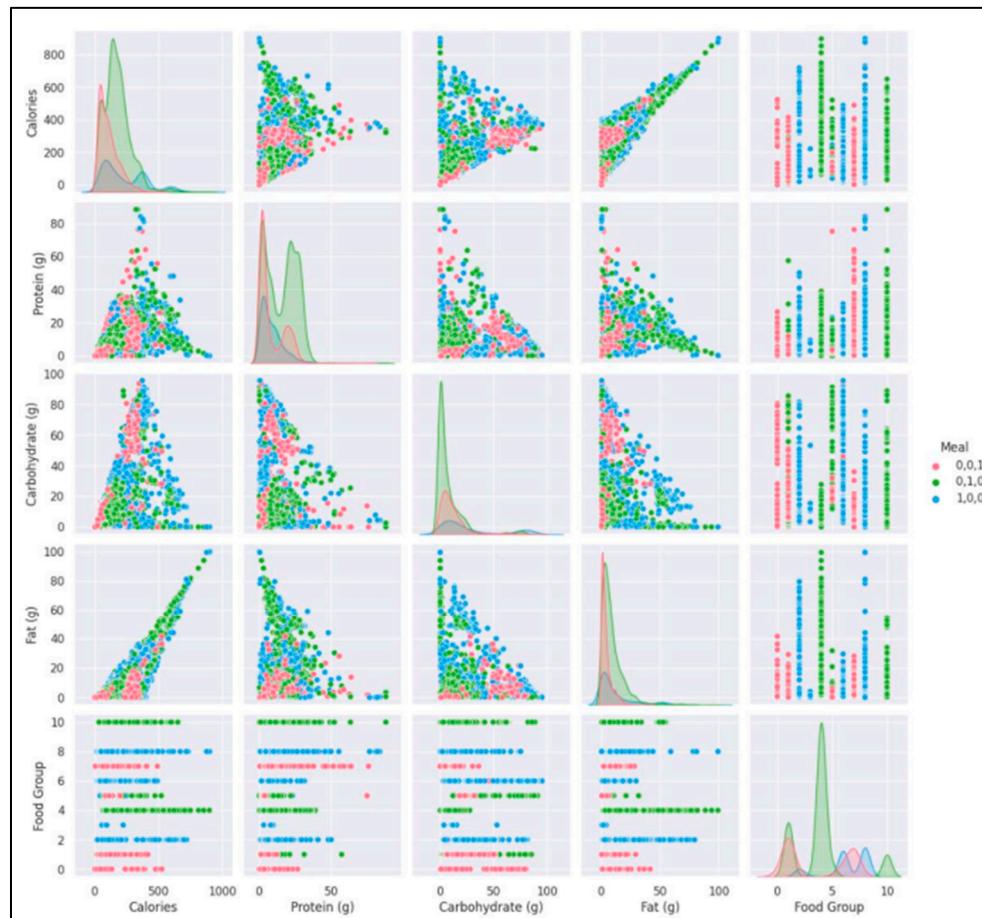


Figure 4. Pair plot for macronutrients and meal without 0,0,0 value.

4. Results and Discussion

The user's calorie goal was computed based on their input details such as height (in cm), weight (in kg), gender, age, and activity level. The activity level was divided into five categories: Minimum (no exercise or desk job), Slightly Active, Moderate, Highly Active, or Extremely Active. Using this information, the Body Mass Index (BMI) was computed as shown below:

$$\text{BMI} = \text{weight}/\text{height}^2 \quad (1)$$

Based on the BMI range, the user was labelled as underweight if below 18.5, normal weight if between 18.5 and 25, overweight if between 25 and 30, and obese if more than 30. The Basal Metabolic Rate (BMR) was computed using the Mifflin–St Jeor equation as given below:

For men:

$$\text{BMR} = (10 \times \text{weight in kg}) + (6.25 \times \text{height in cm}) - (5 \times \text{age in years}) + 5 \quad (2)$$

For women:

$$\text{BMR} = (10 \times \text{weight in kg}) + (6.25 \times \text{height in cm}) - (5 \times \text{age in years}) - 161 \quad (3)$$

The BMR value is then multiplied by the activity factor, which depends on the user's activity level. The activity factor values are 1.2, 1.375, 1.55, 1.725, and 1.9 for Minimum, Slightly Active, Moderate, Active, and Extremely Active categories, respectively. The total daily energy expenditure (TDEE) value is as given below:

$$\text{TDEE} = \text{BMR} \times \text{activity factor} \quad (4)$$

The final TDEE value will be 20% more for a caloric surplus (for underweight individuals), no change if normal weight, 20% less for a caloric deficit (for overweight individuals), or 30% less for obese individuals, as shown below:

For underweight individuals:

$$\text{Final TDEE} = \text{TDEE} + (20\% \times \text{TDEE}) \quad (5)$$

For normal weight individuals:

$$\text{Final TDEE} = \text{TDEE} \quad (6)$$

For overweight individuals:

$$\text{Final TDEE} = \text{TDEE} - (20\% \times \text{TDEE}) \quad (7)$$

For obese individuals:

$$\text{Final TDEE} = \text{TDEE} - (30\% \times \text{TDEE}) \quad (8)$$

After the TDEE value is calculated, the calorie goal is then divided into three parts, namely breakfast, lunch, and dinner. If the user desires a balanced meal, then the calorie goal is divided equally among the three meals. However, if the user has a preference for a particular meal to be heavier, then the calorie goal is adjusted accordingly. For instance, if the user wishes to have a heavier breakfast, then the breakfast calorie goal is set to 40% of the final TDEE value, and the remaining 60% is divided equally between lunch and dinner. Similarly, if the user prefers a heavier lunch, then the lunch calorie goal is set to 40% of the final TDEE value, and the remaining 60% is divided equally between breakfast and dinner. The dinner calorie goal remains unaffected at 30% of the final TDEE value, regardless of the user's preference for other meals. This ensures that the user's calorie goals are customized according to their preferences and requirements and helps in maintaining a balanced and healthy diet.

The calorie goal and macronutrient goal values are calculated using the above formulas and respective meal plans are computed. For example, to give the food for breakfast, only the foods that are within the range of the calorie goal and macronutrient goal are provided. The accuracy of the model was 92%. The class with the label "0,0,0" had the best outcomes, according to the report. Due to the small number of dinner-related items in the train dataset, dinner, or "0,0,1," had the lowest accuracy. In the classes that we are interested in, breakfast, represented as "1,0,0", produced the greatest outcomes. The result of using the random forest classifier is shown in the following Figure 5.

Confusion Matrix				
		0	1	2
True Label	0	2190	1	2
	1	48	340	110
Predicted Label	2	0	84	665
	3	0	75	21
		0	1	2
		0	1	2

Figure 5. Classification analysis.

5. Integration as a Recommending Application

The application was developed using Android Studio, an integrated development environment (IDE) for Android app development. Java programming language was used to code the activity pages while the layout of the pages was designed using XML. The pages included user input fields, drop-down menus, and buttons. The user inputs details such as their name, age, gender, height, weight, and activity level. Based on these details, the application calculates the user's BMI using the standard formula. The user is then classified as underweight, normal weight, overweight, or obese. The application then calculates the Basal Metabolic Rate (BMR) value of the user using the Mifflin–St Jeor equation. The TDEE value is obtained by multiplying the BMR value with the activity level factor. If the user is underweight, the TDEE value is increased by 20%. If the user is of normal weight, the TDEE value remains unmodified. If the user is overweight, the TDEE value is decreased by 20%. Finally, if the user is obese, the TDEE value is decreased by 30%. The resulting TDEE value is displayed to the user. The user can select their preferences such as non-vegetarian or vegetarian diet, and if they would like to have a heavy breakfast, lunch, or a balanced meal. Based on these preferences, the respective calorie value is passed on to the Python function for diet generation.

Flask is a micro web framework written in Python that allows developers to build web applications. It provides a lightweight and flexible framework for developing web applications by providing essential features such as URL routing, templating, and session management. The Flask app contained six endpoints that served as APIs for the mobile app. The endpoints included '/printbf', '/println', and '/printdn' for printing the list of breakfast, lunch, and dinner items for the user, and '/altnbf', '/altnln', and '/altdn' to provide alternatives for the suggested items in the lists. The '/printbf', '/println', and '/printdn' endpoints took arguments for the calorie goal for the respective meal and whether the user preferred a vegetarian or non-vegetarian diet.

The Flask app accessed the food dataset and returned a response containing the three most suitable items that fell within the calorie goal and macronutrient goal of the user for the respective meal. The '/altnbf', '/altnln', and '/altdn' endpoints took an argument for the item name for which the alternative was required. The Flask app then fetched the features of this item from the dataset and calculated the distances and indices of the nearest neighbors using the K-nearest-neighbors algorithm (KNN). It then extracted the nearest neighbors based on similarity in values of the features, i.e., macronutrient amount, and returned the list of nearest neighbors as a response.

The Flask app was hosted on a local server, and the mobile application sent HTTP requests to these endpoints to obtain the required responses. Communication between the Flask app and the mobile app was established using the HTTP protocol, which allowed for easy integration between the two systems. By using Flask, the developers were able to quickly build a backend server that could handle the requests from the mobile app and provide the necessary responses.

The user inputs their necessary details such as calorie goal, non-vegetarian or vegetarian preference, and meal type. The calorie goal is calculated using the Mifflin St-Jeor equation, and the generated value is passed to the function created for the project. The function uses a pre-defined list of food items for each meal type, namely breakfast, lunch, and dinner. Based on the user's calorie goal and macronutrient goal, the function selects three to four food items that best match the input and returns them as a response. If the user wants to view alternatives for the suggested food items, the K-nearest neighbors algorithm is used to create a similarity matrix. The list of alternatives is then generated based on the nearest neighbors found. The generated meal plan, along with alternatives, is displayed to the user through the Android application. The user can also provide feedback on the suggested food items, which is stored in the database and can be used to improve the accuracy of the system in the future. The launch screen and the selections of alternatives are as shown below in Figure 6.

(2)

DietMate

DietMate

Name _____

Age _____

Gender:

Male

Female

Height (in cm) _____

Weight (in kg) _____

Activity level

Sedentary (little to no exercise + work a.. ▾

CALCULATE

(1)

DietMate

DietMate

Sai Prakash _____

21 _____

Gender:

Male

Female

165 _____

55 _____

Activity level

Moderately Active (moderate exercise 3-5.. ▾

CALCULATE

(4)

DietMate

Hi, Sai Prakash

Age: 21

Gender: Male

Your calculated BMI value: 20.2

You are Healthy weight

Your Basal Metabolic Rate(BMR): 1481.2

Your TDEE (in calories): 2295.9

Choose your eating preference:

Non-Vegetarian ▾

Decide your meal split:

Heavy Lunch ▾

VIEW DIET

Figure 6. *Cont.*

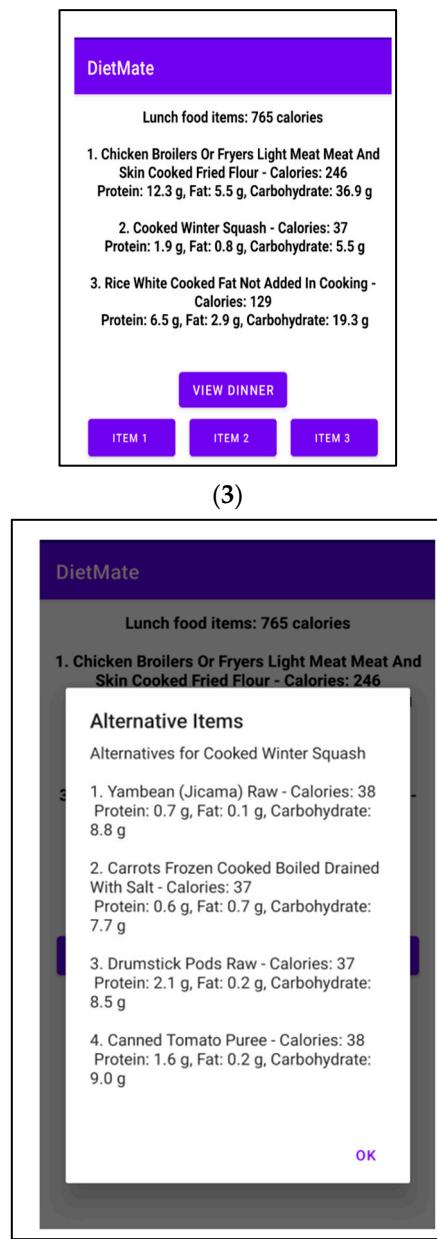


Figure 6. (2) Launch screen. (1) User's input details. (4) User's calorie goal displayed. (3) User's lunch meal plan. (5) Alternatives for suggested lunch items.

6. Conclusions

Maintaining good health and warding off chronic illnesses necessitates the adoption of a nutritious diet. This research work provides a personalized meal plan that adheres to an individual's calorie requirement. In addition to offering a meal plan, it also provides substitutes in case the user needs any. The application developed ensured that dishes were classified according to their meals using a random forest classifier with an accuracy of 91%. Further, to provide substitutes for any dish that was unavailable to the user, the KNN algorithm was adapted. The application can be further enhanced with a bigger data set.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Publicly available datasets were analyzed in this study. This data can be found here: FoodData Central (usda.gov).

Acknowledgments: The author appreciates the contribution made by the undergraduate research team comprising Sai Prakash R and Sai Sriya V for this project.

Conflicts of Interest: The author declares no conflicts of interest.

References

1. Ezugwu, A.E.; Ikotun, A.M.; Oyelade, O.O.; Abualigah, L.; Agushaka, J.O.; Eke, C.I.; Akinyelu, A.A. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Eng. Appl. Artif. Intell.* **2022**, *110*, 104743. [[CrossRef](#)]
2. Aiyar, A.; Rahman, A.; Pingali, P. India's rural transformation and rising obesity burden. *World Dev.* **2021**, *138*, 105258. [[CrossRef](#)]
3. Garcia-Closas, E. The Use of Machine Learning in the Study of Dietary Patterns and Breast Cancer Risk. *Curr. Nutr. Rep.* **2019**, *8*, 365–373.
4. Liu, Y.; Wang, Y.; Zhang, J. New machine learning algorithm: Random Forest. In *Information Computing and Applications: Third International Conference, ICICA 2012, Chengde, China, 14–16 September 2012. Proceedings 3*; Springer: Berlin/Heidelberg, Germany, 2012.
5. Raynor, M.F.; Wing, R.R. Ecological momentary assessment of eating-related behaviors. *J. Nutr. Educ. Behav.* **2013**, *45*, 608–612. [[CrossRef](#)]
6. Mifflin, M.D.; St Jeor, S.T.; Hill, L.A.; Scott, B.J.; Daugherty, S.A.; Koh, Y.O. A new predictive equation for resting energy expenditure in healthy individuals. *Am. J. Clin. Nutr.* **1990**, *51*, 241–247. [[CrossRef](#)] [[PubMed](#)]
7. National Institutes of Health; National Heart, Lung, and Blood Institute. *Disease and Conditions Index: What Are Overweight and Obesity?* National Institutes of Health: Bethesda, MD, USA, 2022.
8. Nguyen, P.H.; Scott, S.; Headley, D.; Singh, N.; Tran, L.M. The double burden of malnutrition in India: Trends and inequalities (2006–2016). *PLoS ONE* **2021**, *16*, e0247856. [[CrossRef](#)] [[PubMed](#)]
9. Osisanwo, F.Y.; Akinsola JE, T.; Awodele, O.; Hinmikaiye, J.O.; Olakanmi, O.; Akinjobi, J. Supervised machine learning algorithms: Classification and comparison. *Int. J. Comput. Trends Technol. (IJCTT)* **2017**, *48*, 128–138.
10. Tran, T.N.T.; Felfernig, A.; Trattner, C. Recommender systems in the healthcare domain: State-of-the- art and research issues. *J. Intell. Inf. Syst.* **2021**, *57*, 171–201. [[CrossRef](#)]
11. World Health Organization. *WHO Factsheet on Obesity and Overweight: World Health Organization Fact Sheet*; World Health Organization: Geneva, Switzerland, 2021.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.