Software System Development (Monsoon2023)

Project Report

Team Name: ***SSD_M2023_30_RONS***
Team No.: ***30***
Project Title: ***Nutrition Counter***
Project No.: ***8***
Instructor/Mentor: ***Dr.Charu Sharma***
GitHub Repository URL: [IAmOnkarSawant/30_NutritionCounter: Nutrition (github.com)](https://github.com/IAmOnkarSawant/30_NutritionCounter)

MembersDetails:
1.Onkar Bharat Sawant(2023201024)
2. Nithin Venugopal(2023201033)
3. Shubham Jaiswal(2023201070)
4. Rhitesh Kumar Singh(2023202013)

Architecture

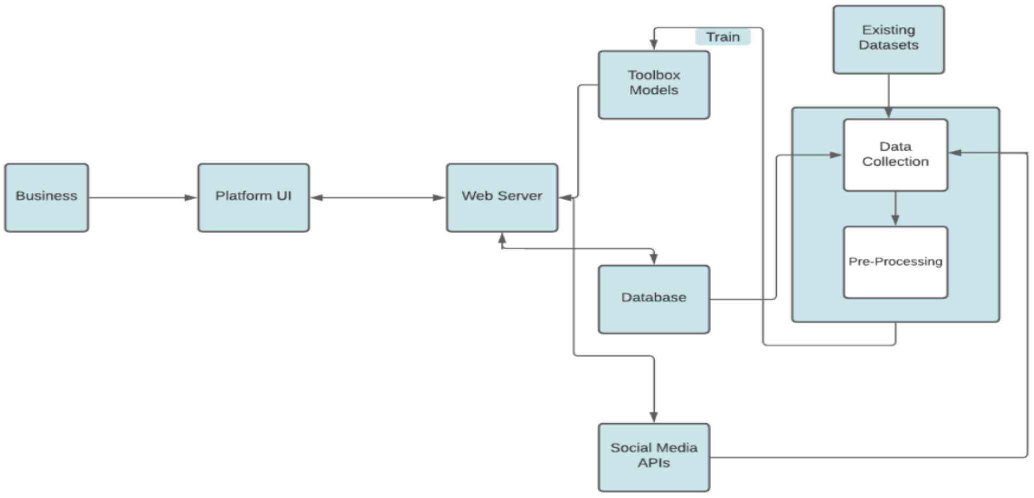

Fig. Architecture diagram

Workflow:

The user can upload the image of the labelled food item either by selecting it from the local device or by capturing the image using the camera. The image will be processed in the backend with the help of a Machine Learning model, extracting text from the image. Recommendations are then made based on the model output and our database.

Logic:

1. DataBase: Why NoSQL?

The user data we receive is unstructured, making NoSQL a more suitable choice over SQL. In our case, we opted for MongoDB.

We have a total of 5 collections named:

Foods: This collection contains ingredients and their respective content values based on various food items we've gathered. The dataset includes around 8000 different food items and their respective ingredients. This information is utilised for making recommendations. The source of this data was Kaggle.
No Cardiac: This collection contains a list of ingredients that are not recommended for individuals with heart diseases.
No Pregnancy: This collection contains a list of ingredients that are not recommended for pregnant and lactating women.
No Children: This collection includes a list of ingredients that are not recommended for children.
No Diabetic: This collection includes a list of ingredients that are not recommended for diabetic patients.

2. Image Upload:

Since the image contains text, anything other than text-based images is not processed by our model. The user uploads the image, which is then stored locally using Multer. The Machine Learning Model (PyTesseract) processes this image. If the input image is "Ingredients text-based," the output image is a list of strings. If the input image is "Table text-based," the output image is a dictionary. In this dictionary, the key is the name of the contents, and the value is either in mg or DV% or both. We make recommendations based on the output of the model.

3. **Model Used:**

Initially, we employed the Keras model for our application. However, we encountered an issue where the Keras model was unable to detect words in a line-by-line format, which was necessary for our use case. For example, in Keras OCR, "Refined Wheat flour" was detected as "Refined," "Wheat," "flour." Despite trying other models, we eventually opted for PyTesseract for our use case. PyTesseract can read the input text line by line, thereby improving the accuracy of the application.

4. **Post Processing:**

After processing the image, we convert the extracted text into JSON format and store the result in a buffer. This JSON-formatted data is then utilised for comparison with our dataset.

5. **Why are we taking the user's Height, Weight, Age and Goal of the user?**

We have two primary purposes:

BMI Calculation:

- Although we are not currently utilising it for recommendations, calculating BMI serves as a foundation for potential future enhancements. This feature can be instrumental in making personalised health recommendations based on BMI and other related parameters.

BMR (Basal Metabolic Rate) Calculation:

- BMR represents the number of calories the body needs to perform essential functions while at rest. It is often expressed in terms of calories per day. To calculate BMR, we employ the Harris-Benedict equations:

For Men:

$$BMR = 88.362+(13.397\times \text{weight in kg})+(4.799\times \text{height in cm})-(5.677\times \text{age in years})$$

For Women:

- $BMR = 447.593 + (9.247 \times \text{weight in kg}) + (3.098 \times \text{height in cm}) - (4.330 \times \text{age in years})$

Once BMR is calculated, we estimate the Total Daily Energy Expenditure (TDEE) based on the activity factor. We assume an activity factor of 1.2 for a sedentary lifestyle:

- $TDEE = BMR \times \text{Activity Factor}$
- Based on the calculated BMR and the user's fitness goal, we generate a sample personalised calorie-based diet plan.

6. **Recommendation:**

The image is categorised as either "Ingredient text-based" or "Table text-based," each requiring a distinct approach for recommendations.

Ingredient Text-Based:

- For images with an "Ingredient Text-Based" nature, we analyse the ingredients list to generate recommendations. Our system maintains a list of ingredients not recommended for specific groups, including Pregnant Women, Cardiac Patients, Diabetic Patients, and Children. Recommendations are tailored based on the potential side effects on these groups.

Table Text-Based:

- In the case of "Table Text-Based" images, our focus is on macronutrients. The Daily Value (DV%) serves as a reference value for nutrients on a food label, based on a daily intake of 2,000 calories—a standard for nutrition labelling. DV% indicates how much a serving contributes to the total daily recommended intake of a nutrient. This reference point is crucial for making recommendations in this context.
- We have used DV% for categorising if a food item with the given DV% is usually recommended for a person on average. If DV% is less than 5% then the macronutrient is considered too low and if the Dv% for a macronutrient is more than 20% then it is considered as too high.

USE CASES:

The use cases for your application include:

Dietary Recommendations:

- Providing personalised dietary recommendations based on the analysis of food images and their nutritional content.
- Recommending suitable food items for specific health conditions, such as for pregnant women, cardiac patients, diabetic patients, and children.

Macro-Nutrient Analysis:

- Analysing macro-nutrients in food items based on text extracted from images.
- Calculating and presenting Daily Value (DV%) information to users for a better understanding of nutritional content.

Health Goal Planning:

- Calculating BMI (Body Mass Index) and BMR (Basal Metabolic Rate) for users.
- Estimating Total Daily Energy Expenditure (TDEE) based on BMR and activity levels.
- Providing sample personalised calorie-based diet plans aligned with users' fitness goals.

Image Processing and Text Extraction:

- Accepting images of labelled food items, either uploaded or captured using the camera.
- Processing images through a Machine Learning model (PyTesseract) to extract text information.
- Differentiating between "Ingredient text-based" and "Table text-based" images for relevant recommendations.

Category-Specific Recommendations:

- Offering category-specific recommendations for different types of input images.
- Tailoring recommendations based on the nature of the text extracted, such as ingredient lists or table formats.

Future Scope:

- Collecting and utilising data related to BMI and other health parameters for potential future recommendations.
- Expanding the application's capabilities to encompass additional health-related insights and advice.

These use cases collectively contribute to the overall goal of providing users with personalised dietary guidance and promoting better health choices.