SMART DIET: AN INTELLIGENT MOBILE APPLICATION TO ASSIST DIET CONTROL USING ARTIFICIAL INTELLIGENCE AND COMPUTER VISION

Wenyan Zhu[1], Yu Sun[2]

[1]Sage Hill School, 20402 Newport Coast Dr, Newport Beach, CA 92657
[2]California State Polytechnic University, Pomona, CA, 91768, Irvine, CA 92620

ABSTRACT

Studies have shown the importance of diet in correlation with obesity and several chronic diseases. Trying to reduce the incidence of diet related diseases, we designed a mobile application for users to keep track of their nutritional intake and thus promote healthier eating patterns. We implemented a deep learning model into the application that can make predictions when given an image and analyze the nutrients for that food item. The sum of daily nutritional information is displayed to users on the dashboard, as well as a letter grade to help visualize their progress on healthy eating. Every past diet log is saved locally on Shared Preferences for the users to pull up as needed. The users have full control over how to use the application, and it is designed to raise awareness of how much nutrients are suggested daily in comparison to each individual consumption. We evaluated the effectiveness of the application with an experiment to test out its accuracy, and the results supported the application's potential as well as inspired ideas for future improvements.

KEYWORDS

Artificial Intelligence, Nutrients, Mobile APP

1. INTRODUCTION

Nutrients are chemical substances required by all lives to produce energy, and humans acquire necessary nutrients through nutrition, or food intake. The consumption of a healthy diet on a regular basis can provide numerous benefits, including protection against heart disease and diabetes, support of muscle and bones, and improvement of the immune system [1]. Incorporating healthy diets into one's lifestyle can help prevent and fight off chronic diseases, as well [2]. On the other hand, consumption of an unbalanced diet can lead to a myriad of problems, such as type 2 diabetes, obesity, hypertension and cardiovascular disease, or certain types of cancers [3].

First, a poor diet can directly lead to obesity, which is a risk factor for many other diseases. From 2011, the percentage of adults aged 18 years and older who have obesity in California has increased from 23.8 to 30.3 in 2020, and one of the explanations behind this increase is diet [5]. The main causes of obesity is a high intake of sugar and fat, processed and fast food, and a low amount of exercise, so the energy consumed will be stored in the body as fat [4]. As a result of obesity, one will have a higher risk of adopting other diseases, such as heart disease, respiratory

David C. Wyld et al. (Eds): AIAA, DPPR, DSA, ICSS, IOTE, NLPTA, WEST, CACIT, BMLI - 2022
pp. 141-152, 2022. CS & IT - CSCP 2022 DOI: 10.5121/csit.2022.122311

diseases (asthma and sleep apnea), skeletal issues (osteoarthritis), gallstones, and a variety of thirteen cancers [6].

Next, in relationship with heart diseases, a diet that affects the functioning of the heart or elevates the blood pressure within the body can be problematic. Consumption that is high in saturated fats, trans fat, and cholesterol can increase the risk of heart conditions, such as atherosclerosis, particularly referring to the buildup (plague) of fats and cholesterol in the artery walls; this causes arteries to narrow and restrict proper blood flow [7]. A high consumption of sodium can cause an increase in blood pressure, thus disrupting the homeostasis of the body and causing improper regulation of cardiac output.

Lastly, diets play an essential role in the contribution or cause of Type 2 Diabetes. Unlike Type 1 Diabetes that is caused by an autoimmune response of the body, Type 2 Diabetes is closely associated with lifestyle, such as obesity or eating too much sugar and fat [8]. Therefore, the treatment of Type 2 Diabetes also involves diet regulation: to consume lower levels of fat and higher levels of fiber [12].

The ones mentioned above are not the only diseases with correlation to diets, and in fact, nutrition can have an impact on every system of the body. Considering the effect of macronutrients and micronutrients, a poor diet can often lead to chronic diseases, subsequent complications, and eventually death, making it a topic that deserves attention from people.

Both governmental agencies and private organizations have initiated programs to target nutrition problems and promote health equity. For example, CDC has created the ECE Obesity Prevention and High Obesity County Program to specifically target the issue of obesity. Other programs of CDC, including IMMPaCT, REACH, and SPAN, focus on issues of micronutrient intake, chronic diseases management, and physical activity level respectively [9]. However, these programs mainly fund other local organizations usually located in areas where obesity or certain disease prevalence is high to promote healthy eating. This can be limited in scale and overlook many people in need who are not living in those areas, and qualifications to engage in the programs can be complex. In addition, the effectiveness of those programs are hard to measure because they are expansive and not directly interacting with people who need assistance. Thus, it creates many uncertainties to how well the program is running if there is a large dataset coming from a variety of locations.

Another method for people to promote nutritional health is through a nutritionist. This can be effective for people who need a very specialized diet with many restrictions and needs, while at the same time being able to monitor their health. Associations such as the American Nutrition Association are also training health professionals and practitioners to maximize the benefits of personalized nutrition [10]. On the other hand, this method has a practical problem: some people are unable to afford the cost of frequent nutritional check-ups. This is in alignment with the economic ladder and the health wellbeing gradient, which increases at every social class, stating how every American is healthier than those with lower income levels [11]. Therefore, for people in poverty who are afflicted with obesity or chronic diseases that need a nutritionist, such as heart diseases or diabetes, they do not have access to such treatment.

In this study, we developed a mobile application which integrates a deep learning model that will act as a personal nutritionist through the use of computer vision. Deep learning is a type of neural network with three or more layers that gather information and learn from it, similar to a human brain. The artificial intelligence learns from the data through repeatable loops to increase accuracy, and then it is able to make predictions when given new data. Computer vision refers to the process the AI extracts data from images in order to train itself, using CNN, or Convolutional

Neural Network. When given an image, the AI is able to identify a pattern and convolve the data, then going through max pooling to simplify the data. The two steps are repeated until reaching a satisfactory result, then the data will be flattened and full connection will take place. Through this process, the AI is able to learn when it is given data.

Our model is aware of thus far 30 basic foods to identify within uploaded pictures. We trained it by providing a large amount of images per food, and the images were scraped from flickr with use of their image API. As such, it has consumed many high definition pictures and can consistently identify foods within images. The users can also use the built-in camera from their device to take images of the food and the application is able to classify and localize the food, then providing the nutritional information of the food. By saving the data of each meal and compiling them together, the application shows the daily nutritional intake of seven pieces of nutrients as well as caloric level on the home page. A letter grade will also appear on the dashboard to inform the users on how healthy their diets are. Furthermore, this application is convenient to use at any time by simply taking a photo of a meal, and users are able to search up previous meals to see past diet logs as well.

The rest of the paper is organized as follows: Section 2 gives the details on the challenges that we met during the experiment and designing the sample; Section 3 focuses on the details of our solutions corresponding to the challenges that we mentioned in Section 2; Section 4 presents the relevant details about the experiment we did, following by presenting the related work in Section 5. Finally, Section 6 gives the conclusion remarks, as well as pointing out the future work of this project.

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. How Should the User Interact with this App

Being an application that closely tracks a user's daily lifestyle, it is difficult to consider how to make it efficient and attractive at the same time. There are a variety of ways one can design an application, and it requires comprehensive planning to find the right approach. The application should be convenient to use and easy to navigate, while at the same time providing practical feedback on the user's diet. Based on those characteristics, a considerable time needs to be spent to determine how to maximize users' experience when they interact with the application. Additionally, since the ideal use case for this app is a long-term commitment (upwards of at least a month of daily use) it is important that users are able to use the app without getting annoyed or confused in the first couple days of using the app. Therefore, it is hard to consider what the most important part of the application is and how to provide a satisfactory experience even with a simple design.

2.2. How do we Make the AI

This application can only be as effective as the artificial intelligence that is put into it. It is important for the artificial intelligence to be both robust and accurate while also making sure that developing the artificial intelligence does not take up too much development time. Thus, to increase the effectiveness of the AI, particular food items that are more commonly consumed should be taught to the AI first. However, it is difficult to determine exactly how many foods should be included in the system as well as which foods they should be due to the wide variety of foods in the world. If one attempts to make the AI so effective as to recognize hundreds of food items, one also needs to spend a lengthy period of time to build the application. Because designing the UI and training the system to analyze various foods also take up developing time, it is impractical to include that many food items. Thus, a balance must be found to make the AI robust and the developing time efficient.

2.3. How Should we Store Data

Consideration must be put into how data is collected from the user, since data must constantly be either written or read by the application. This is an important technical problem to consider in order for the application to run, and every method to store and collect the information has its pros and cons. For example, two methods we can consider are cloud storage and local storage, each having different advantages and disadvantages. Cloud storage is tied to the internet and requires building a sufficient backend, though it can hold more data; local storage alternatively cannot store as much but can be quicker than relying on internet speeds. When the method is considered, one must pay careful thought about what information from the user is necessary for the application. Because privacy can be a sensitive topic, and the application does not need to store data that is not necessary, setting up data storage can be a tedious task.

3. SOLUTION

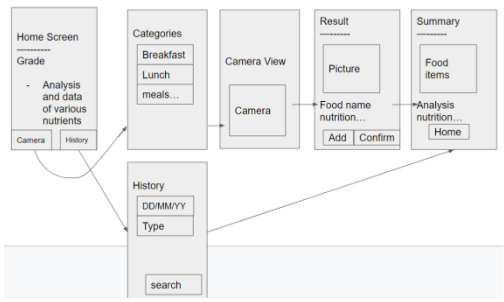

Figure 1. Overview of the solution

This application logs users' diets and provides feedback in return using image classification and a neural network called Deep Learning. The AI is trained by giving it large amounts of data and allowing it to learn through repeatable loops, and it is able to make accurate predictions based on the training once it has learned from varied images. The AI uses computer vision to extract the data from given images to train itself by identifying a pattern and convolving the data. Through convolution and max pooling, the AI would be able to predict new data more accurately. When users use the application, they would select the category of meal they are consuming, and then take pictures or upload them at the camera page. The AI will be able to detect and recognize food items from the meal using image classification. The nutritional information of each food item is given to the AI, and this information would be pulled up from a dictionary of food information and listed on the result page for the appropriate food. Then, the application gives users the option to upload more pictures of their meal until they are satisfied and proceed to the summary page. The AI would add all the information in this meal together to show the users their total nutritional intake. The data of each meal is saved in Shared Preferences locally to allow the users to search up their past meals at any time. To obtain a grade, the users' average daily intake of each nutrient is divided by the recommended level of intake, and if the result reaches a certain level, the users would see a higher grade appearing on their dashboard. Their total nutritional intake is also reflected on the dashboard below the grade.

```
void _imageSelection() async {
  var imageFile = await ImagePicker().getImage(source: ImageSource.gallery).
  then((value) {
    if (value != null)
    {
      _imageClasification(value!);
    }
  });
}

void _cameraSelection() async {
  var imageFile = await ImagePicker().getImage(source: ImageSource.camera).
  then((value) {
    if (value != null)
      {
        _imageClasification(value!);
      }
  });
}

void _imageClasification(PickedFile image) async {
  var output = await Tflite.runModelOnImage(
    path: image.path,
    numResults: 2,
    threshold: 0.1,
    imageMean: 0,
    imageStd: 255,
  ).
  then((value) {
    setState(() {
      if (value == null) print("did not successfully load");
      print(value);
      _listResult = value;
      processImage(image);
      foodInfo = getCorrespondingFood(value!);
    });
  }
  );
}
```

Figure 2. Screenshot of code 1

The code above explains the process in which the user gives the AI and image, either from his camera or gallery, and the AI would analyze it can give back a result. The analysis of the image happens when the user presses a particular button to process in the application, and after the process is completed, the AI's prediction would be displayed on the result page. The result would be shown on the application for the user, but it is not included in the code.

To provide the AI with an image, the user has two options. The first is _imageSelection, which is when the user chooses to have an image uploaded from their image gallery. The second is _cameraSelection, which is when the user chooses to have an image taken from their camera instead. The user will be prompted to take a picture using their phone's built-in camera application, and the result will be given to the AI.

The AI uses a method named _imageClassification to predict the image, which includes a tensorflow model that trains the AI. If the AI can successfully predict the food item from the image, it will display the image from the user as well as its prediction in the form of a simple String. To show more information such as nutritional content, getCorresponding Food will add the other information on the screen, which is again displayed to the user.

- Code Context
  - This is the code where the phone gives the AI an image (camera or gallery) and the AI can make a decision
  - A user must press a button in order to have this occur
  - The AI's prediction is displayed on the page for the user but is not shown here
- Code Explanation
  - _imageSelection
    - occurs if the user chooses to have an image from their image gallery
  - _cameraSelection
    - occurs if the user chooses to have an image from their camera feed instead. The user is prompted to take a picture on their phone's camera app and the result is given to the AI
  - _imageClassification

■ has the tensorflow model (the AI) try and predict the image.
■ if it can successfully predict the image…
■ display the image the user sent
■ the AI's prediction is in the form of a String (a simple label). To show the user more information such as nutritional content, getCorrespondingFood is used to turn the prediction into plentiful data. This data is again displayed to the user.

```
String generateMealName() {
  String todaysDate = DateFormat("MM-dd-yyyy").format(DateTime.now());
  return todaysDate + " " + RecordedData.typeOfMeal;
}
Future<void> _saveName() async {
  final prefs = await SharedPreferences.getInstance();
  String foodNameList = generateMealName();
  List<String> names = <String>[];
  for(FoodInfo f in RecordedData.foods){
    names.add(f.name);
  }
  setState(() {
    prefs.setStringList(foodNameList, names);
  });
  if (prefs.containsKey(foodNameList)) {
    print("Confirmed " + foodNameList);
  }
}

Future<void> _saveMeal() async{
  await _saveName();
  MealData m = RecordedData.generateMealData();
  TotalIntakeData.addInfoToday(DateFormat("yyyy-MM-dd").format(DateTime.now()), m);
  Navigator.push(
    context,
    MaterialPageRoute(builder: (context) => MealInfo(mealID: generateMealName())),
  );
}
```

Figure 3. Screenshot of code 2

The code above explains the process in which the meal results are saved to the user's phone's drive, and it takes place after the user has uploaded all images from their meal and likes to see the meal results. This step is necessary because it allows the user to look up the information later for their own purposes at any time. This code produces a specifically formatted name and saves the meal contents in a specific way that makes it easier to be stored. A plugin named Shared Preferences was implemented for data storage, which allows the saving of basic data to the phone's hard drive. Because it is an example of local storage, it maximizes the convenience for users to pull up past meals. About the code, the String generateMealName will generate a name that the meal contents will be, which will be used for the saving process. Then, save Name creates a list of the food names that are different from the foods themselves to make the data smaller. This is important because Shared Preferences can only save simple data. The contents of the list can be reverse-lookup later to make sure no data is lost during the saving process. Lastly, this list is saved into Shared Preferences under the name generated by generateMealName.

● Code Context
✧ This code runs whenever the user is finished generating their meal and would like to see the meal results.
✧ Before we show them the meal results, we have to save their meal into their phone's drive first, in case they would like to view a summary of it later on.
✧ This code will produce a specifically formatted name and save the meal contents in a specific way so that its easier to save.
✧ We save with a plugin called Shared Preferences, which saves basic data to the phone's hard drive
● Code Explanation
✧ generateMealName
■ generates a name that the meal contents will be
✧ saveName
■ generates a meal name using generateMealName
■ creates a list of the food names, not the foods themselves

- ◆ this is to make data smaller, Shared Preferences can only save simple data
- ◆ we can reverse-lookup the contents of the list later so no data is lost when saving a meal
- ◆ saves the list into Shared Preferences under the generated name

```
double calculateGrade () {
  double r_calories = 2000;
  double r_carbs = 200;
  double r_sodium = 1500;
  double r_cholesterol = 300;
  double r_fat = 60;
  double r_potassium = 3000;
  double r_protein = 80;
  double r_vc = 100;

  double calorieGrade = cal/r_calories;
  double carbGrade = carbs/r_carbs;
  double sodiumGrade = sod/r_sodium;
  double cholesterolGrade = chol/r_cholesterol;
  double fatGrade = fat/r_fat;
  double potassiumGrade = pot/r_potassium;
  double proteinGrade = pro/r_protein;
  double vcGrade = vc/r_vc;

  print("calorie grade: " + calorieGrade.toString());
  print("carbohydrate grade: " + carbGrade.toString());
  print("fat grade: " + fatGrade.toString());
  print("protein grade: " + proteinGrade.toString());
  print("sodium grade: " + sodiumGrade.toString());
  print("potassium grade: " + potassiumGrade.toString());
  print("cholesterol grade: " + cholesterolGrade.toString());
  print("vc grade: " + vcGrade.toString());


  double totalGrade = calorieGrade + carbGrade + sodiumGrade + cholesterolGrade + fatGrade + potassiumGrade;
  double averageGrade = totalGrade/8;
  print("average grade: " + averageGrade.toString());
  return averageGrade;
}

String getLetterGrade(double grade)
{
  if (grade > 0.9) return "A - Wow! You are eating awesome!";
  else if (grade > 0.75) return "B - Great! You are just a step away!";
  else if (grade > 0.5) return "C - Keep going! I believe in you!";
  else if (grade > 0.3) return "D - You are on the way there!";
  else return "F - Hey, everything starts from a poor diet!";
}
```

Figure 4. Screenshot of code 3

```
averageCalories() async {
  final prefs = await SharedPreferences.getInstance();
  if (prefs.getDouble("Average Calories") != null)
  {
    cal = prefs.getDouble("Average Calories")!;

    print(cal);
    setState(() {});
  }
  else{
    cal = 0;
  }
}
```

Figure 5. Screenshot of code 4

The code above is part of the analytics screen, or the home screen, where the user's statistics are displayed to them. The purpose of this code is to calculate a grade based on how much the user's diets are reaching certain threshold points. The code calculateGrade will create a digit as a reference for the letter grade. First, the values for the recommended daily amounts were obtained and setted up. Then, the user's average intake for each nutrient is divided over the recommended amount for each piece of food information. Usually this will yield a number between 0 to 1, which getLetterGrade will use this number to generate a letter grade from F to A. As a result, a letter grade is acquired based on the total grade calculated in calculateGrade. Only the letter grade will be shown to the user to improve clarity, alongside with an encouraging phrase, and the total grade calculated will not be shown to the user.

- ● Code Context
- ✧ This is on the analytics screen (home screen) where the user's stats are displayed
- ✧ this calculates a grade based on how they've been eating per meal
- ● Code Explanation
- ✧ calculateGrade
- ■ We first set up the values for the recommended daily amounts.

- ■ We then divide the user's average intake by the recommended amount for each piece of food information.
- ■ We take an average of all sub-grades to find our total grade.
- ✧ GetLetter Grade
- ■ generates a letter grade from F to A based on the total grade calculated in calculateGrade
- ■ this letter grade is shown to the user, not their total grade.
- ✧ AverageCalories
- ■ we grab the average amount of calories by looking at SharedPreferences under the corresponding name
- ■ this data is initially saved in the results screen when the user finishes a meal
- ■ this process is identical for the other 7 pieces of information.

4. EXPERIMENT

Since the users are using the application to log their diets, it is crucial that the application can correctly identify the food items in the images that were uploaded. The functionality of the application thus depends on the accuracy of the AI, so the experiment was designed revolving around this concern. The results will verify whether or not the AI can work sufficiently, and using this data, areas of improvements can be identified to further progress the development of the intelligence. To carry out the experiment, thirty images for each food were given to the AI for it to identify, which sums up to a total of 450 images. A yes or no answer was recorded to indicate whether the AI successfully identified the food or not, if not, the falsely identified food item was also recorded. The confidence level is taken note of for the purpose to determine nuance.

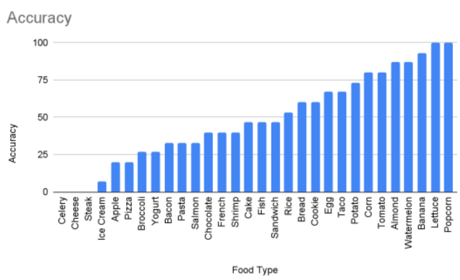

Figure 6. Accuracy vs. food type

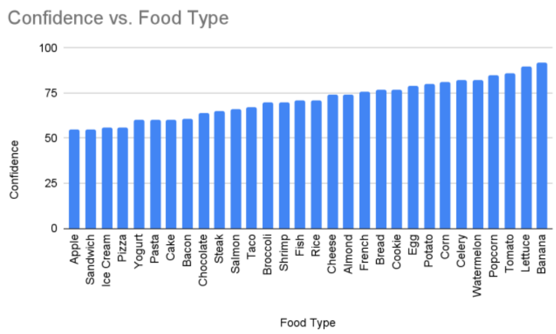

Figure 7. Confidence vs. food type

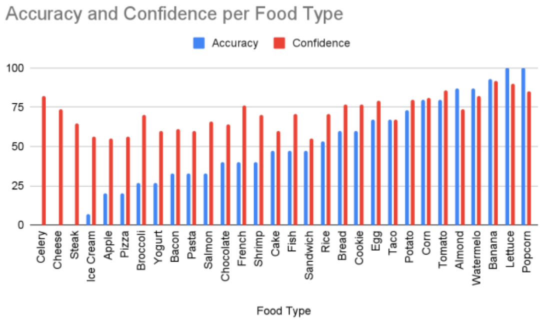

Figure 8. Accuracy and confidence per food type

The general accuracy was determined by dividing the amount right for a given food by the amount of images tested per food. This fraction was then represented as a percentage. The least accurate predictions were a tie between cheese, steak, and celery, at 0%. The most accurate answers were a tie between lettuce and popcorn, at 100%. The foods closest to 50% accuracy were a tie between cake, fish, and sandwich, at 47%, and rice, at 53%. Out of the 30 foods tested, 13 of them were over 50% accurate, therefore approximately 43% of the foods were sufficiently accurate. The average accuracy between all 30 foods was 48.93%.

The average confidence was determined by calculating the mean confidence for all predictions per given food. The lowest average confidence was that of an apple and a sandwich, at 55% confidence. The highest average confidence was that of a banana, at 92% confidence. Since all foods had over 50% confidence, we consider them sufficiently confident in their predictions. The average confidence between all 30 foods was 71.40%.

The experiment conducted addresses the above challenges as it gives us a good indicator as to what the AI is good at detecting and what it cannot sufficiently predict accurately. Foods above the 50% accuracy threshold (with some exception) were generally simpler foods such as potatoes and popcorn, as compared to foods such as steak or ice cream. These foods are likely easier to predict for the AI because they do not typically vary from item to item in real life . Such explains why fruits and vegetables such as tomatoes, lettuce, and bananas scored high but foods with multiple ingredients scored lower. Foods such as pizza or steak or ice cream, for example, may have been difficult to predict due to the plentiful myriad of ways that such items are prepared in the real world. Such variance may throw off the AI's pattern recognition and is either a sign that the AI cannot sufficiently pick up on multiple patterns, or that our food selection needs to be more precise and rigid. As shown by the graph depicting accuracy and confidence, there is no correlation between confidence and accuracy.

5. RELATED WORK

Researchers conducted an experiment testing the effectiveness of a food benefit program that uses several financial methods trying to promote more healthy eating [13]. The participants are low income people, who are grouped into different testing conditions, one with incentives to purchase fruits and vegetables, one with restriction to purchasing sweetened food, one with both incentive and restriction, and the last one a control with no policy implemented. As a result, the weekly purchase of fruits increased most significantly when both the incentive and restriction are used, which supports that a food program paired with incentive and restriction can promote customers to buy more healthy food. This program is similar toHealthyou in that they both encourage people to be aware of their health and food choices. Since shopping for groceries is a common necessity, it frequently offers the participants opportunities to improve in their diet, which is similar to the Healthyou that helps the user on a daily basis. On the other hand, there are many differences between this program and the application, one is the participants, two is the method, and three is the proactiveness. The food program is effective because it chooses low income people as the participants, who are more likely to be attracted to the incentives being offered. Healthyou, in contrast, is open to any users who want to utilize the application to their own advantage, and will only be as useful as the users want it to be. The methods differ because the food program is more systematically organized and specifically targeted at the low income people, as stated in the research paper that "data for the present study were collected as part of a randomized trial that enrolled lower-income adults". Healthyou, though, is again more general and uses the same model for every user with interest. It also allows the user to have more freedom over their diet and health as they can use the application to their will.

This research paper summarizes the findings of an analysis on the Special Supplemental Nutrition Program designed for women, infants, and children (WIC) [14]. The program provides supplemental healthy food and nutrition education for the participants; the experiment analyzes the relationship between participant's duration in the program with household food insecurity (HIF), child diet quality on a given day, and child obesity around 60 months. In the end, the experiment supports that longer participation results in a reduced HFI and higher diet quality, but there is an unexpected rise of obesity levels, which they would further investigate into. The WIC program is an effective method that attempts to improve people's dietary health as well as their awareness, and it is similar to Healthyou in that the length of participation has a positive effect on the result. The longer the user uses the application to keep track of their nutrition, supposedly they will be able to gain a better dietary health. The WIC program is different from Healthyou in that it is very specific, not only with the participants, but also the factors that are examined and methods to analyze them: "Outcomes assessed…were HFI (USDA 6-item Household Food Security Screener), child diet quality on a given day [Healthy Eating Index (HEI)-2015], and obesity (CDC BMI-for-age ≥95th percentile)". On the other hand, Healthyou keeps track of single nutritional information to analyze only the overall diet quality, which can be more comprehensible but less diverse.

The effect of a worksite health promotion program is measured by examining employees' levels of physical activity, weight loss, and in this research paper diets [15]. Eligible programs must be "worksite-based health promotion intervention with minimum study duration of eight weeks", which have the aim to improve employees' physical health. For each program, one or more dietary data is obtained in order to analyze the effectiveness of each program. Sixteen programs end up being compared to one another, focused either on education or environment, and they generally support a positive trend in the consumption of fruits, vegetables, and fat intake. One thing that the worksite health intervention program has in common with the application is keeping the record of fat intake. Healthyou is capable of showing the fat consumption of each food item, each meal, as well as the accumulation on a given day, and it is able to do so for six other nutrients. One thing that differs between the two projects is that the intervention program has a large socioeconomic purpose behind the enterprise. As noted in the paper that "obese people also suffer more sickness and absences from work", improving the physical health of workers could likely yield better profit for a workforce. In contrast, Healthyou has no correlation with profit or economy while helping the users to be more aware of their dietary health.

6. CONCLUSIONS

To address certain health issues and aid with disease prevention, a mobile application named Healthyou, which integrates a deep learning model, is designed that acts as a personal nutritionist. By raising the users' awareness of their nutritional intake and allowing them to visualize their diets, Healthyou aims to ultimately reduce the incidence of heart diseases, Type 2 Diabetes, and general obesity. In order to achieve this, the AI must have the ability to interpret and present nutritional data of foods and relate the information to a standard amount. Consequently, it stresses on the importance of the AI to extract valid information from the uploaded or taken photos, as well as a robust algorithm for calculations. Thus, an experiment was designed to test out the AI's ability. The AI received 15 sample images for each of the thirty food items it learned beforehand, and a yes/no answer was recorded for AI accuracy. If the AI did not recognize the true identity of the food item, the falsely identified food was also recorded. Confidence level of the AI was recorded for each identification because when the AI falsely identifies something, it is helpful to determine whether it cannot recognize the item or mistakenly confused it with another. From the result, conclusion can be drawn that the AI is capable of correctly recognizing food items, especially with simpler foods, which it accurately identifies the food with a higher confidence. However, improvements can be made for the AI to recognize more complex foods such as sandwiches and pizzas, which had inaccurate readings with moderate confidence levels during the experiment. Nonetheless, the experiment supports that Healthyou can assist people's dietary health, and it would be more effective once received further training and modification.

While Healthyou has potential, there are several current limitations that hold back the AI's full potential to make a greater impact. For one thing, the AI is limited in food classification choice as its knowledge is limited to thirty so far, while there are millions of food present in the world. To accommodate users' varied diets addressing their dietary patterns and backgrounds, the AI would be more practical if it can recognize more foods because that will produce more accurate statistics about the diets. However, this limitation persists because new choices of food must be added and trained manually. Secondly, the training data for the AI is currently scraped from flickr, which is not a viable long term choice because not all obtained images are relevant or appropriate for training purposes. As a result, the training would go poorly because the AI is misleadingly viewing irrelevant images as true visual representation of a certain food item. This also ties into the third limitation, considering that the training data was also sourced from flickr. The issue of irrelevant and inappropriate images occurs again, and as the experiment shows, the AI needs to receive more valid training in order for most of its predictions to be sufficiently accurate.

The first limitation that needs to be addressed currently is accuracy. To increase the accuracy for more complex food items, the AI needs to be trained with more legitimate images, and it would be best to have a better source than flickr. Once the AI can distinguish the subtle differences between similar foods, such as between celery and lettuce that posed a problem during the experiment, from the new training, the limitation is sufficiently eliminated. Then, new food items can be added to the AI to expand its practically for keeping a more detailed and precise diet log.

REFERENCES

[1] World Health Organization. A healthy diet sustainably produced: information sheet. No. WHO/NMH/NHD/18.12. World Health Organization, 2018.

[2] Roberts, Christian K., and R. James Barnard. "Effects of exercise and diet on chronic disease." Journal of applied physiology 98.1 (2005): 3-30.

[3] Lazar, Mitchell A. "How obesity causes diabetes: not a tall tale." Science 307.5708 (2005): 373-375.

[4] Fock, Kwong Ming, and Joan Khoo. "Diet and exercise in management of obesity and overweight." Journal of gastroenterology and hepatology 28 (2013): 59-63.

[5] Puckett, Mary, et al. "Nutrition and physical activity strategies for cancer prevention in current National Comprehensive Cancer Control Program plans." Journal of community health 41.5 (2016): 1013-1020.

[6] Wolin, Kathleen Y., Kenneth Carson, and Graham A. Colditz. "Obesity and cancer." The oncologist 15.6 (2010): 556-565.

[7] Vasan, Ramachandran S., et al. "Arteriosclerosis, atherosclerosis, and cardiovascular health: Joint relations to the incidence of cardiovascular disease." Hypertension 78.5 (2021): 1232-1240.

[8] Atkinson, Mark A., George S. Eisenbarth, and Aaron W. Michels. "Type 1 diabetes." The Lancet 383.9911 (2014): 69-82.

[9] Easier, Making Healthy Eating. "CDC's Division of Nutrition, Physical Activity, and Obesity." (2021).

[10] Basiotis, P. Peter, Carol S. Kramer-LeBlanc, and Eileen T. Kennedy. "Maintaining Nutrition Security." Family Economics and Nutrition Review 11 (1998): 4.

[11] Khullar, Dhruv, and Dave A. Chokshi. "Health, income, & poverty: Where we are & what could help." Health Affairs 10 (2018).

[12] Barnard, Neal D., et al. "A low-fat vegan diet and a conventional diabetes diet in the treatment of type 2 diabetes: a randomized, controlled, 74-wk clinical trial." The American journal of clinical nutrition 89.5 (2009): 1588S-1596S.

[13] Cohen, Alicia J., et al. "Increasing use of a healthy food incentive: a waiting room intervention among low-income patients." American journal of preventive medicine 52.2 (2017): 154-162.

[14] Anderson, Christopher E., et al. "Longer Special Supplemental Nutrition Program for Women, Infants, and Children (WIC) Participation Duration Is Associated with Higher Diet Quality at Age 5 Years." The Journal of Nutrition 152.8 (2022): 1974-1982.

[15] Ni Mhurchu, Cliona, Louise M. Aston, and Susan A. Jebb. "Effects of worksite health promotion interventions on employee diets: a systematic review." BMC public health 10.1 (2010): 1-7.