A Web Application of Food Recognition and Nutrition Visualization

KehanYang ky2398, Boyu Yang by2267
Dept. of Electrical Engineering, The Fu Foundation School of Engineering and Applied Science
Columbia University
ky2398@columbia.edu, by2267@columbia.edu

Abstract—**People share their life and delicious food every day in social networking platform, such as Facebook, Instagram and so on. Meanwhile, they pay great attention on their health, telling health apps what they eat and how much they eat every day. This project designs a web application of food recognition based on convolutional neural network, and analyzes the nutrition information of each category with visualization using images and diagrams, and finally recommending users with scientific diet plan. Food-11 dataset is used for classification model training and web visualization performance evaluation.**

Keywords-food recognition; web application; nutrition; visualization

I. Introduction

Nowadays, as people's living quality is getting better and better, we care more and more about the food combination we eat and the nutrition we get every day. More and more people require not only the diversity of food but also a healthy eating habit. Therefore, it is necessary and important to help people understand the nutrition facts of the food they frequently eat and choose which kind of food they would like to eat.

This project develops a website application to recommend food for users. In order to provide proper recommendation of food types for users, we need to first get the information of the users' recent eating habits. In recent years as social networks develop rapidly, more and more people are accustomed to post recent food photos on their personal accounts, so it is a natural idea that following users' habits and using these images as eating habit information. We use a convolutional neural network to do image classification to judge the types of food. A food type-nutrition table is established for evaluating nutrition users recently get from food. Different images and diagrams for visualization of nutrition and food recommendations are implemented for users. We use Food-11 dataset [1] for the training process of our neural network and the experiment of data visualization and food recommendation results. This application can be used in various areas such as social network and health care application development, living and eating habit evaluation and so on.

II. Related Works

Herranz, Min and Jiang utilized external knowledge, visual and contextual information from food image to construct an intelligent food analysis system and applications, based on cross-modeling and CNN [2]. SRI developed a technology recognizing and analyzing food using images for estimating the food portion and nutrition [3]. Bossard, Guillaumin and Gool introduced a new way to automatically recognizing food categories utilizing Random Forests to mine discriminative parts of the image pixels [4]. Hassannejad implemented a deep CNN architecture, which was InceptionV3 developed by Google to recognize images from Food 101 dataset, achieving 88.28% top-1 accuracy [5]. A method of introducing vertical layer structure capture of food image is proposed in [6], where Martinel, Foresti and Micheloni achieved 90.27% top-1 accuracy based on wide-slice residual networks for food recognition.

III. System Overview

Our web application can be divided into three major parts, which are pre-trained neural network model, front end and back end. A LeNet-5 neural network [7] is trained before our web runs. Front end is responsible for showing the comprehensive functions of our platform, instructing users step by step and revealing the recognition result and detailed information of the food. Back end provides support for the front end, burdening the task of feeding images forward into the network, obtaining the type result and computing nutrition ratio.

A. Dataset

The dataset we use is Food-11 dataset. It totally contains 16,643 food images, which are divided into three parts. Training dataset includes 9,866 images, validation dataset includes 3,430 images and evaluation dataset includes 3,347 images. There are 11 food categories, which are Bread, Dairy product, Dessert, Egg, Fried food, Meat, Noodles/Pasta, Rice, Seafood, Soup, and Vegetable/Fruit. The total size of the dataset is about 1.16 GB [1].

Fig.1 Food-11 Dataset

B. Syster Flowchart

The first step of the whole system is training classification neural network using the dataset. After training process, we will get a classification model. On web page, users can upload different food photos, and this model will be used to do classifications on these images. According to the nutrition table we created and the classification results, the system will search for the nutrition of the current food types. Finally the nutrition will be presented by different images and diagrams and users will get different food recommendations full of nutrition they currently need.

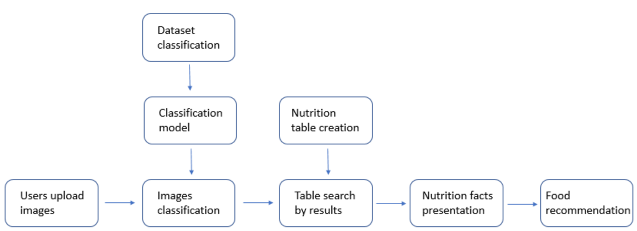

Fig.2 System Flowchart

IV. Algorithm

A. Image Classification

In this project, we use LeNet-5 [7] as our classification network structure. LeNet-5 is a convolutional neural network structure, which was first trained on the MINIST dataset. LeNet-5 only contains two convolutional layers, two pooling layers and one fully connected layer, which is shown below in figure 3. In order to decrease the overfitting, we add one batch normalization layer after every pooling layer.

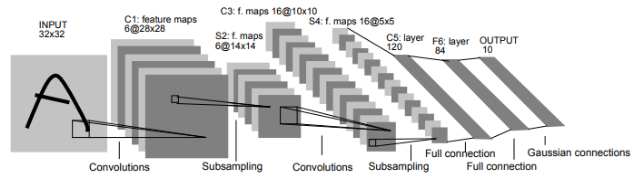

Fig.3 Structure of LeNet-5 [7]

We use Google Cloud Platform as our training platform and TensorFlow Keras as our deep learning framework. The environment is Ubuntu with two 7.5GB CPUs and one Tesla K80 NVIDIA GPU.

B. Web Page Implementation

We built a web application for food recognition on Food-11, enabling model loading, images uploading, food category recognition and nutrition analysis.

This implementation relies on several tools to realize the interaction between user pages and supporting programs. Flask is a framework for python to render dynamic web pages, providing useful APIs for fetching and transferring arguments between web pages and python functions. For our web page arrangement and content design, html is the programming language, and Bootstrap, CSS and D3.js are very helpful tools to generate more complete and beautiful web pages. Figure 4 shows relationship between web pages.

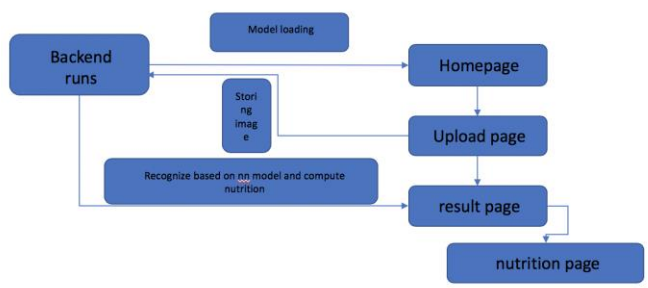

Fig.4 Website Structure

We totally have 4 pages, containing the home page with brief introduction about our website and a button linking to the page for uploading the user images. In the uploading page, once a user finishes the task of uploading images, he/she can click on the recognition button and wait for the recognition to complete. After this completes, the result page will show a full result with original food image, category result as well as a link for detailed nutrition information page in every row for every image. And in the right part inside these rows, user can change the amount of every food to see different overall nutrition analysis and our recommendation in the bottom of the page. And our default value for every food in every image is 50g. Figure 5~9 show different contents in our web application.

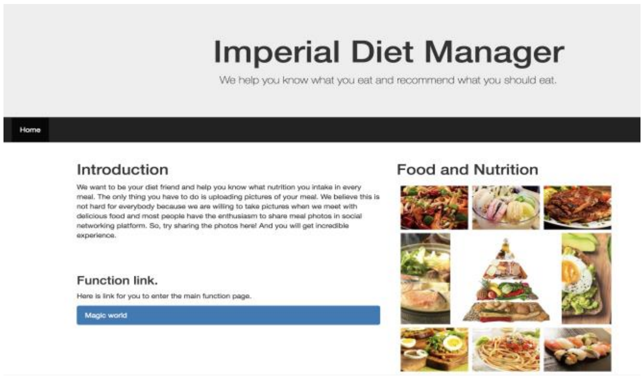

Fig.5 Home Page

Fig.6 Uploading Page

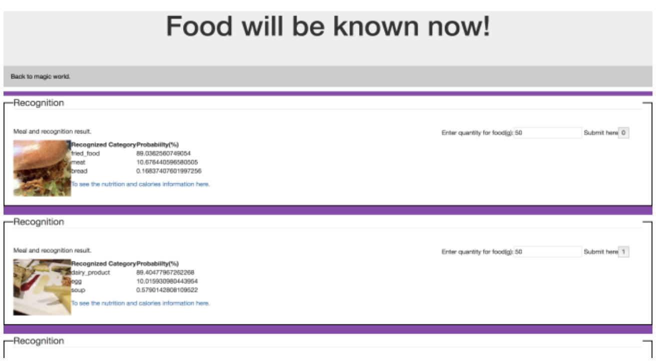

Fig.7 Recognition Result

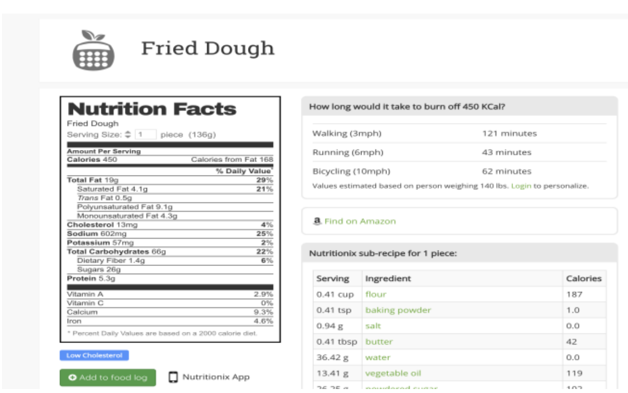

Fig.8 Nutrition Information Page

Fig.9 Overall nutrition analysis

V. Software Package Description

A. *classification.ipynb*

This jupyter notebook file is used for classification model training. Because of the limitation of memory, we use image generators to generate batches of images. Then we use TensorFlow Keras to establish network structure and train our model. During training process, we save the weights of model with best validation accuracy. When making predictions of single image, we reload the weights and use the model for food recognition.

```
model = Sequential()
model.add(Conv2D(16, kernel_size=(5, 5),
                 strides=(1, 1),
                 activation='relu',
                 input_shape=input_shape))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(BatchNormalization())
model.add(Conv2D(32, (5, 5), activation='relu'))
model.add(MaxPooling2D(pool_size=(2, 2)))
model.add(BatchNormalization())
model.add(Flatten())
model.add(Dense(150, activation='relu'))
model.add(Dense(num_classes,
activation='softmax'))
```

Code Snippet 1. Network Structure

B. *Web Page Package*

All of the file used for running the website is contained in a directory named webserver. People who wants to run the application should use command to enter in the webserver directory and activate related environment, which allows flask, Keras and TensorFlow to take effect, and run the server.py. For our macOS environment, we show a group of local command of starting the program below.

Fig.10 starting the server.py

After loading the neural network model is completed, there will be a hint saying: “Running on http://0.0.0.0:8111/”. Just copy this link to any browser. Once user enter the home page, he/she can click on the “Magic world” button in the left bottom of the page and choose image files to upload in the uploading page. Uploading process can be processed in a fast speed. Clicking on “recognition now” will take people to the result page. In every row of the recognition result. People can seek for more detailed information through the nutrition link, shown in the red circle of the following figure. And the amount of the food in one image can be changed through the right column in every row of one food, which is circled in blue in figure 11.

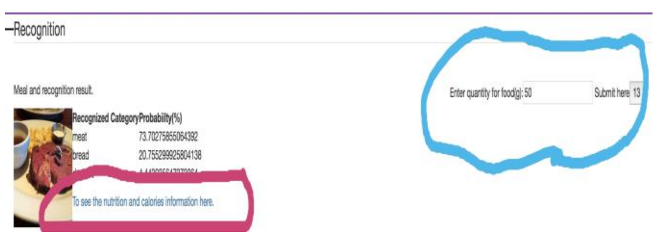

Fig.11 Nutrition Information and Change Amount

After one try of recognition food images, users can click on the “back to magic world” on the top of the result page to go back to the upload page. The button is shown below in figure 12.

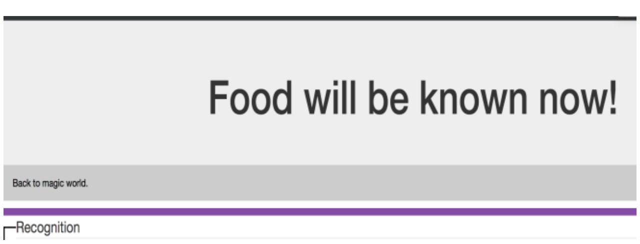

Fig.12 Back to Magic World Button

VI. Experiment Results

For food recognition, because of the limitation of resources and time, we are restricted to do enough image pre-processing operations as well as build more complicated network structure and we finally get an accuracy about 38% on evaluation dataset for the 11 food categories. However, it is enough for us to finish the following part of this project and show the practicability and application probability. In further works, we can implement several image pre-processing algorithms and adjust the network structure to get a better result. For web page implementation, the performance of visualization is various and understanding.

VII. Conclusion

Through the designing of a complete web application for food recognition, the authors have implemented neural network construction, modeling saving, modeling loading, and user interface arrangement based on tools such as TensorFlow Keras, flask, html, Bootstrap, CSS and D3.js.

After we improved the network structure by introducing dropout and batch normalization, we reached an accuracy of about 38% on the evaluation dataset. We still need to improve the accuracy by several methods. Some food category may be always falsely judged to another kind, due to the limited class number of our dataset and the complexity of the food recognition problem itself. Food images have similarity in the same category but also contain variations among the same group. We will use larger dataset with more explicit classification in the future and do experiment on algorithms automatically analyzing the ingredients and nutrition of the food from its image.

In conclusion, this web application offers a great start for combining sharing image function and recognizing food as well as analyzing the nutrition. And we will have a greater improvement in the future.

Boyu Yang is responsible for the construction and training of the convolutional neural network. Kehan Yang is responsible for designing the responsive web page.

Acknowledgment

THE AUTHORS WOULD LIKE TO THANK HTTPS://WWW.W3SCHOOLS.COM/BOOTSTRAP/BOOTSTRAP_GET_STARTED.ASP [8]FOR THE OFFERING OF GREAT INSTRUCTIONS IN BOOTSTRAP AND HTTPS://WWW.NUTRITIONIX.COM/ [9]FOR USING THEIR NUTRITION PAGE.

References

[1] Food-11 dataset, https://mmspg.epfl.ch/food-image-datasets.

[2] Herranz, L., Min, W., & Jiang, S. (2018). Food recognition and recipe analysis: integrating visual content, context and external knowledge. arXiv preprint arXiv:1801.07239.

[3] https://www.sri.com/engage/products-solutions/food-recognition-technology

[4] Bossard, L., Guillaumin, M., & Van Gool, L. (2014, September). Food-101–mining discriminative components with random forests. In

European Conference on Computer Vision(pp. 446-461). Springer, Cham.

[5] Hassannejad, H., Matrella, G., Ciampolini, P., De Munari, I., Mordonini, M., & Cagnoni, S. (2016, October). Food image recognition using very deep convolutional networks. In *Proceedings of the 2nd International Workshop on Multimedia Assisted Dietary Management* (pp. 41-49). ACM.

[6] Martinel, N., Foresti, G. L., & Micheloni, C. (2018, March). Wide-slice residual networks for food recognition. In Applications of Computer Vision (WACV), 2018 IEEE Winter Conference on (pp. 567-576). IEEE.

[7] Lecun, Y. L. , Bottou, L. , Bengio, Y. , & Haffner, P. . (1998). Gradient-based learning applied to document recognition. Proceedings of the IEEE, 86(11), 2278-2324.

[8] https://www.w3schools.com/bootstrap/bootstrap_get_started.asp

[9] https://www.nutritionix.com/