

NuTriX: A smart way to recommend your Foods

Name1,

School of Computer Engineering and Electrical

Pennsylvania State University

University Park, PA 16802

Email: <http://www.michaelshell.org/contact.html>

Name2

School of Computer Engineering and Electrical

Pennsylvania State University

University Park, PA 16802

Email: <http://www.michaelshell.org/contact.html>

Abstract—This project searches for a manner in which to predict the dietary restrictions within a food based off an ingredient list. Intuitively, some dietary restrictions are identifiable without question. For example, cheese will trigger a flag for lactose intolerance. The additional step that this project takes is identifying the probability of other dietary restrictions being triggered as a result of correlation between ingredients. To extend the aforementioned example, our machine might detect that a recipe with cheese might also have high sodium content, triggering high blood pressure as a dietary restriction.

I. INTRODUCTION

Food plays an important role in our every life; it is essential for the human body to function. People have different preferable food depending on various conditions such as allergens, or congenital disease. Therefore, the food recommendation system is one of the important research areas. There is adequate research related to food recommendation and machine learning, as there is more public data available than before.

The recent research on food recommendations is various. Mostly, researchers were able to predict food's ingredients by taking a picture. They used various machine learning techniques such as SVM, multiclass- Adaboost, etc. One notable work is, "Food-101 – Mining Discriminative Components with Random Forests", which relies on Random Forests Discriminative Components to classify foods into 101 categories. Researchers put emphasis on their use of super pixels, comparable to mosaic tiles, as a stand in for image patches, as they claim are best for object detection. The RFDC model performed well on their own testing data at 50.76% accuracy, only being outperformed by Convolution Neural Network by about 6%. However, when tested on MIT-Indoor data at 58.36%, it was outperformed by Discriminative Mode Seeking and Intuitionistic Fuzzy Values by over 2-8%. With even more fruitful results, "Automatic Chinese Food Identification and Quantity Estimation" relies heavily on sparse coding and SVM to classify foods into 50 categories. Researchers did not compare their models to others developed in the field, but rather compared different variations of their models against each other. They found that Multi-class Adaboost with Top-5 accuracy proved to be the most accurate, with an accuracy of 90.9%. However, the average accuracy of their work is 68.3%.

In this paper, we target to develop a system that can classify food by its ingredients from pictures and texts. We aim to

create two machine learning models; first is picture to text model, second is text to text model. Each model will be tested on a different machine learning model and compared to see which one performs better.

II. DATASETS

Our study relies on the data retrieved from an online machine learning platform, kaggle.com. The website contains free datasets published by its users, mainly used for data science purposes. We three datasets; the first two were found on kaggle.com, the third dataset was found on eightportions.com. The first dataset, ingredient v1.csv was uploaded by the company Datafiniti under the CC BY-NC-SA 4.0 copyright. Datafiniti compiled 10,000 entries containing attributes such as "brand, name, manufacturer, category, features, and more" (features being the ingredients list). The second dataset was created by Elisa for the purpose of creating a food recommendation system. The dataset was collected from allrecipes.com; it contains about 50,000 recipes, including its name, ingredients, image_url, cooking_direction, nutrition, and id. The third dataset was created by Ryan Lee under the ODC Attribution license. Lee scraped this dataset from various websites; it contains about 125,000 recipes, including name, ingredients, instructions, source, and picture.

In essence, the required data needed in each dataset was an ingredients list for each food entry, which were provided in both of the files collected from Kaggle. As aforementioned, our first dataset contained 10,000 entries and the second one contained 50,000. With that being said, a considerable amount of the entries contained null values for certain values, which was the first obstacle encountered during the project. An even bigger issue that was encountered was that a considerable amount of entries were non-food items (such as Clorox), causing noise in the data.

III. PREPROCESSING

The preprocessing of data proved to be the most extensive part of the process; the steps taken have been broken down into four parts below:

A. Cleaning Data

As a result of the data being sourced through opportunistic methods rather than experimentally, missing and inappropriate

data proved to be a cumbersome obstacle. For ingredients v1.csv, there were some ingredients that contain special characters such as $\hat{}$; some of the food was left as none. It might affect the accuracy of the model so first, we removed special characters and none from the dataset. For core-data_recipe.csv, we neglected some columns because the dataset was too big. The columns that were removed were image_url, cooking ingredients, and nutrition.]

B. Classifying Dietary Restrictions Feature Selections

We created a dictionary of allergens that indicate each dietary restriction and ingredients that contain them. We aimed to perform a machine learning model on following nine different dietary restrictions; lactose intolerance, celiac, diabetes, high blood pressure, allergens, halal, kosher, vegetarian, and vegan. Then, we performed a string check on each dataset to detect allergens. For example, when we looked at the ingredients of 'Potato Bacon Pizza.', we detected that there was cheese. Thus, we added lactose intolerance into its allergens column. Referencing back to the issue of non-food data entries mentioned in "Overview of Datasets", one more round of data clearing was applied after classifying dietary restrictions. As a fix to the issue, data entries that contained no allergens were removed, since non-food items would not raise a flag on any of the dietary restrictions.

C. Feature Scaling

D. One Hot Encoding

Since the ingredients column contained entries of lists. The data was therefore three dimensional. To lower the array's dimensionality back down to two, one hot encoding was performed. As a result, each possible ingredient was then listed as its own column and each row displayed a tally of how many times said ingredient shows up in the recipe.

E. Separating into Feature Vectors and Target Variables

After One Hot Encoding the ingredients, one more step remained before feeding our data into the models - the separation of variables. The target variable was an array of the different dietary restrictions that was looped through, allowing us to apply the model on each one.

IV. TRAINING AND TESTING DATA

One of the challenges faced when searching for correlations of each individual dietary restriction was that the models only identified the correlations for one target variable at a time. However, the process of training and testing the model based off of each restriction followed identical form without loss of generality. Therefore, a for each loop was applied on the set of dietary restrictions, calculating the correlations for each dietary restriction one at a time.

V. POSTPROCESSING AND VISUALIZATION

VI. RESULTS

VII. CONCLUSION

The conclusion goes here.

ACKNOWLEDGMENT

The authors would like to thank...

REFERENCES

- [1] H. Kopka and P. W. Daly, *A Guide to L^AT_EX*, 3rd ed. Harlow, England: Addison-Wesley, 1999.