CooNet: A New Recipe Creation Website with Data Visualization

Jiawei Wu[1], Siya Xie[2], Ting Wang[3], Yanxiang Zhou[1], Yuebai Gao[1]

1 H. Milton Stewart School of Industrial and Systems Engineering
2 School of Biological Sciences
3 School of Civil and Environmental Engineering

Introduction and motivation

Flavor pairing is not only an essence of culinary practice, but also a wisdom. Although rooted in personal experience, the booming of cutting-edge technologies has transformed this ancient subject into an emerging research field, Gastrophysics.1 Several recipe or meal plan searching systems have been developed, such as Yummly, Food.com, and Appetit. However, most of them focus on food and nutrition-oriented data but overlooked the importance of customer-oriented data source. Moreover, hardly any platform aims at providing inspiration on new recipe creation, where rational food pairing and bridging methods are required. Therefore, there is a demand for an interactive recipe recommendation and new recipe creation platform for cooks or cooking lovers, which can pair food ingredients based on both nutrition information and customer rating information, thus providing ideas for new recipe creation.

Problem Definition

To build an interactive web application for new recipe creation, three main steps should be achieved: 1) prepare the recipe database that contains information like ingredients, nutrition, and customer ratings of each recipe; 2) build up a machine learning model to predict the score of different ingredient combination based on the historical rating data; 3) visualize the data and recipe creation process in an interactive network.

In the current project, we developed a state-of-art website to realize recipe exploration and innovation. Specifically, we utilized a xgboost regression machine learning model based on the customer ratings of 230,000 + recipes to predict the score of food ingredient combinations, which could guide ingredient recommendation and new recipe creation. The user-selected and system-recommended food ingredients are visualized in an interactive network enabled by JavaScript d3, where the user can create a new recipe step by step. Existing top recipes are recommended, and the detailed cooking information is provided, so that the user can get more inspirations on cooking methods and steps for newly created recipes.

Proposed Method

Our approach features several innovations. Firstly, the customer rating data of the recipes, which directly reflect if combining several ingredients is tasty or not, is fully utilized to guide the ingredient recommendation and recipe creation. Secondly, we use regression ML models to dynamically predict the score of the newly created recipe. Thirdly, the recipe tags are used to split the training recipes into several clusters, thus creating several different ML models to achieve more precise prediction. Finally, the ingredient network realizes data visualization, thus enabling interactive and intuitive recipe creation process.

Back-end: data processing and predictive model

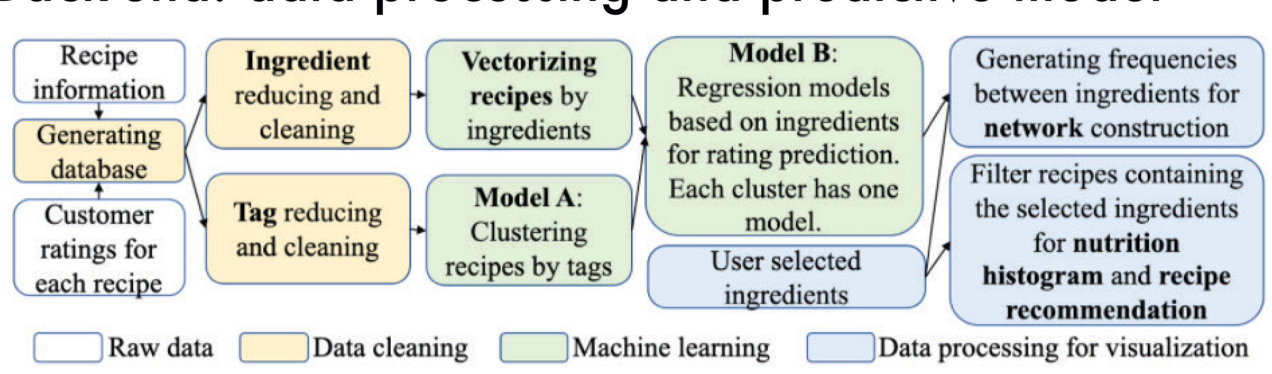
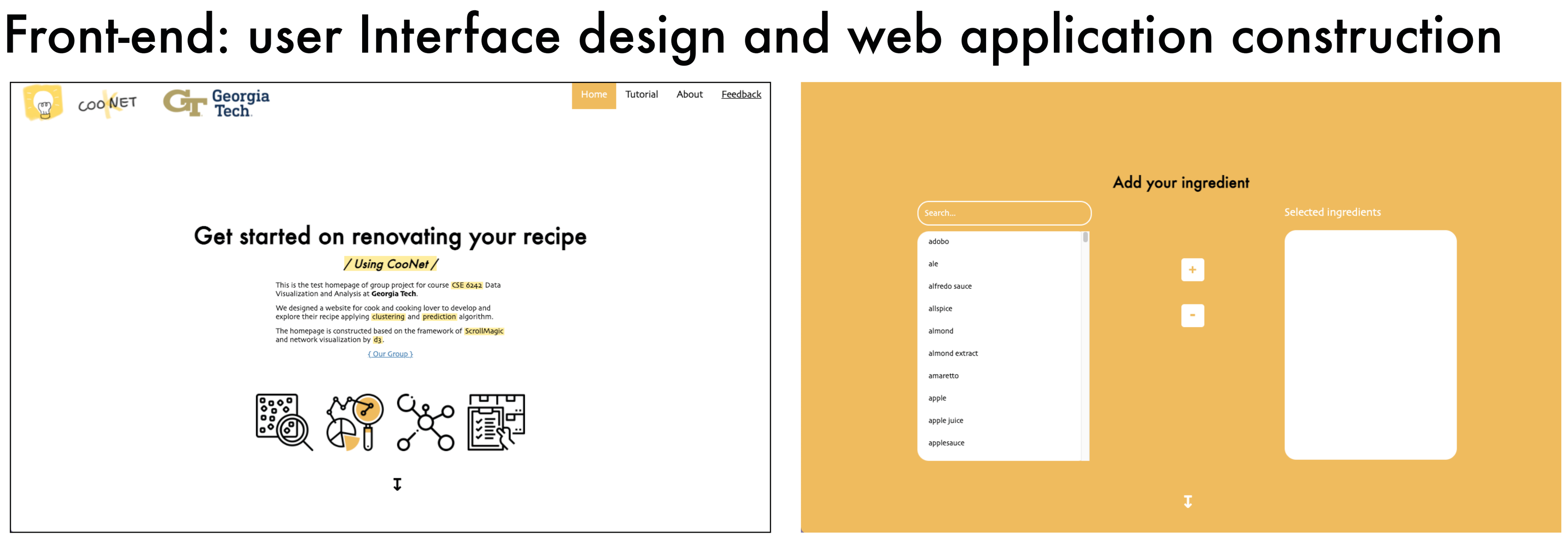


Figure 1.
Flow chart of back-end data processing procedures.

Our data is a Kaggle dataset crawled from Food.com (GeniusKitchen) online recipe aggregator. We used Python for general data cleaning and processing. The database was constructed by combining recipe information (300 MB) and customer ratings (350 MB). The attributes we used are recipe tags, ingredients, nutrition, cooking steps, and ratings. Our back-end model contains two parts. Model A is a clustering model based on the recipe tags. Tags are used to narrow down the range of recipes used for recommendation. To avoid the situation where the number of recipes meet all tags is too small, we did not directly use tags as filters. Instead, we built a cluster model, our Model A, to partition all recipes into a few clusters based on their tags and assign a cluster to the user selected tags.

Model B is a group of similar regression models based on recipe ingredients, which is used to predict the score of ingredient combinations for ingredient recommendation. The recipes were vectorized based on 450 ingredients, so that each recipe was represented as a 450-dimension vector. The output to be predict is the recipe rating. The model training and testing was carried out using Python and sklearn. We tested several candidate models for the regressor: xgboost, SVM, Random Forest, ridge regression, LASSO, linear regression. Determined from model performance on the test data, we eventually chose xgboost.

Front-end: user Interface design and web application construction



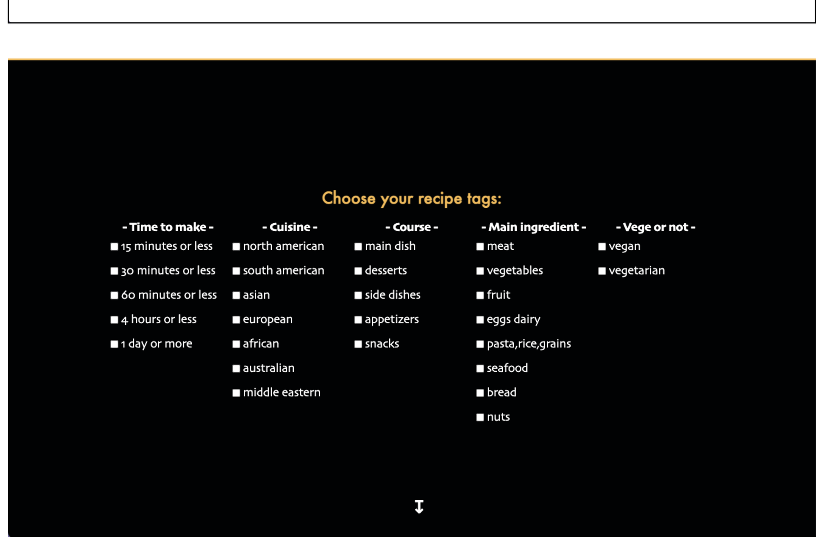


Figure 2. Screen shots of CooNet user interface. (a) Welcome page. (b) Ingredient list. (c) Recipe tags. (d) Result page.

The homepage consists of four sections, welcome, ingredient list, recipe tags, and start. The first section introduces our website to users, including the core idea of our project and primary coding tools applied, together with our affiliation and website usage (Figure 2).

The controller of the web application was implemented using the python application, Flask. In Flask, we created various routes to enable passing data between frontend and backend. After the user chose the ingredients, tags and pressed the start button, user input data would be passed from frontend to backend using the AJAX jQuery. Then the controller would call python functions to compute data required for the network plots, the histograms and the recommended recipes displayed on the result page. The controller would wait until the data are computed before passing them to the frontend with Flask route. The data for network plots would be passed to network.js, and the data for histograms and recommended recipes would be passed to histogram.js.

Heroku was used to publish the web application. Since the memory quota is limited to 100mb, we slide the dataset into 3 files and read them one after another to prevent exceeding memory quota.

Experiments and Evaluation

Description of testbed

The evaluation will be centered on whether the product accomplished the project's objectives as stated in the introduction. Hence, the main questions will be answered by experiments are:

- What are the functionalities of CooNet?
- How accurate is the rating prediction made by the regression algorithm?
- How is the user experience of the CooNet?

Overview of CooNet functionalities

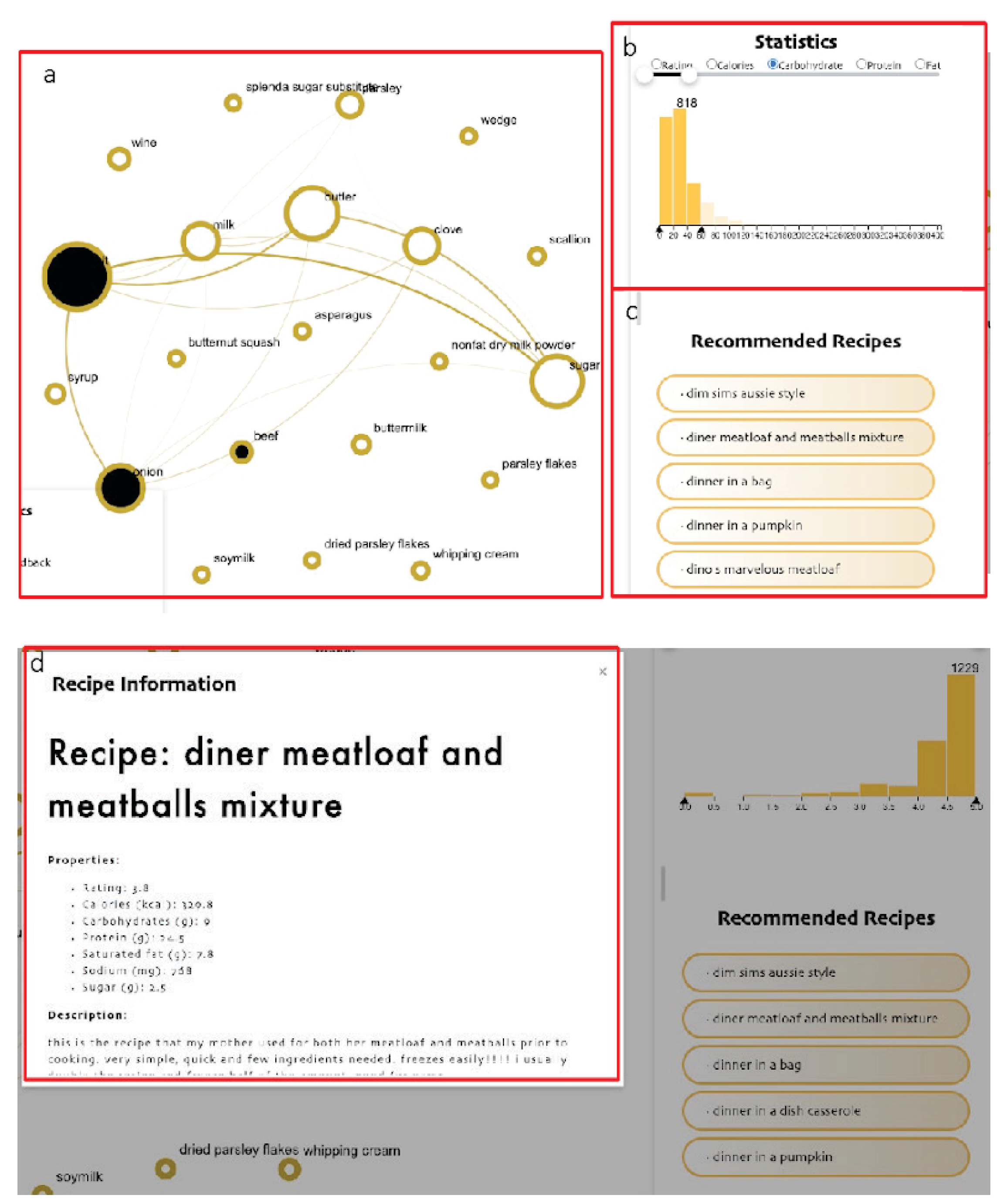


Figure 3. The result page. (a) Ingredient network. (b) Statistics graph. (c) List of recommended recipes. (d) Pot up window of the detailed information of the recommended recipe.

After several food ingredients and recipe tags are selected, a food ingredient network is generated (Figure 3a). The black nodes are 3 user selected ingredients at the main page, and white nodes are the regression model recommended top 20 ingredients that go well with the user selected ingredients. The statistics (Figure 3b) shows the rating and nutrition histogram of all recipes that containing all selected ingredients (black ones in the network). Here, the user can select different items and drag the scale to narrow down the range. The Recommended Recipes shows the top 5 recipes that fall into the user specified range in the statistics graph (Figure 3c). Clicking on one recipe, a window will pop up, showing the detailed information of that recipe (Figure 3d). By clicking on one of the white nodes in the network, that ingredient will be added to the selected ingredients (becoming black), and the new recommended ingredients, statistics, and recommended recipes will be updated.

Evaluation of prediction algorithm

$$RMSE = \sqrt{\frac{\sum_{i=1}^{test} \left(p^{(i)} - y_{test}^{(i)}\right)^2}{N_{test}}} \qquad (1)$$

To determine the accuracy of our algorithm, we will randomly divide the recipe dataset into two parts, 80% of recipes will be used as training dataset and 20% as test dataset. From our experiment, xgboost gives the uniformly lowest RMSE among all the model candidates, around 0.9, and the normalized RMSE is about 0.18.

Evaluation of user experience

We designed a user experience survey to find out how well our web app CooNet meets user's needs. The questions aim to evaluate users' degree of satisfaction to the functionalities, interface design, and recipe recommendation results of CooNet. We have received 26 responses as of 12/4. The detailed results are shown in Appendix I. About 53.8%, 34.6% and 11.5% users rated 5, 4, and 3 scores on the overall functionalities of the web app, indicating that our functionalities still need improvement. The user interface gained 73.1% full score, showing that the interface is user friendly. Although 61.5% users are fully satisfied with the recommendation results, there are still 7.7% users rated score of 2. Overall, users are generally satisfied with CooNet, but the functionalities and recommendation algorithms need further improvements.

Current limitations and how to improve

1. We will combine data from multiple sources to make a larger database. Combining data from multiple sources can increase the database size and make the models more accurate.
2. We can further improve the machine learning models. To improve that, we can reduce the vector dimension by aggregating the ingredients or find more suitable regression models for sparse matrix.
3. The web interface could be further improved.

Conclusion and Discussion

In this project, we successfully constructed a web application, CooNet, for recipe recommendation and creation. We achieved the three main goals for this work. First, we collected and processed the raw data, and generated the database for machine learning and data visualization. Then, we built a clustering model based on recipe tags to split the recipes into groups, and built regression models for each group to predict the score of food ingredient combinations. In the end, we visualized the user selected and model recommended ingredients in a network. Improvements in database size, predicative algorithm, and user interface of the web application as well as the functionality and stability of the connection API are of requirements based on our user feedback investigation.