



Construction of Support System for Demand Driven Design of Cocktail Recipes by Deep Learning

Soichiro Ota^{1(✉)}, Kohei Otake^{2(✉)}, and Takashi Namatame^{3(✉)}

¹ Graduate School of Science and Engineering, Chuo University,
Hachioji 112-8551, Japan
a14.dxte@g.chuo-u.ac.jp

² Department of Information and Communication Technology, Tokai University,
Shibuya City 108-8619, Japan
otake@tsc.u-tokai.ac.jp

³ Faculty of Science and Engineering, Chuo University, Hachioji 112-8551, Japan
nama@indsys.chuo-u.ac.jp

Abstract. Cooking recipes have become available by various ways. However, there are not always recipes that can satisfy any request. In order to reliably provide recipes that can meet his or her needs, it is necessary to newly produce recipes that meet requirements as needed. In addition, a support system is necessary for people who do not have much knowledge to easily devise their favorite recipe. We propose a decision support system for demand driven design of cocktail recipes, that is systemized using Deep Learning due to diversity of ingredients and combinations, differences in taste, etc.

Keywords: Cocktail recipe · Deep learning · Deep neural network

1 Introduction

Cooking recipes have become available by various means. However, despite the huge number of recipes, there are not always dishes that perfectly match each individual's preference. Also, even if they exist, it is not always easy to find them. It is also clear that too many choices can cause problems [6]. For example, it can be a cause of inhibition of motivation to the selection, a decrease of satisfaction to the selected object, and so on. Likewise, because of huge combination of cocktails made from multiple liquors, beverages, etc., there are not always recipes that can satisfy any request. In order to reliably provide recipes that can meet his or her needs, it is necessary to newly produce recipes that meet requirements as needed. In this study, we focus on cocktails where recipes are simpler than general cuisine and devise a method to generate recipes based on requests. Since combinations of ingredients are complicated, we make a support system by deep learning. A support system is necessary for people who do not have much knowledge to easily devise their favorite recipe.

2 Dataset

We used 3110 recipes obtained from the website [5] that posted cocktail recipes. The variables contain the quantities of 645 different ingredients (treated as different ingredients if units of quantity are different) (Table 1).

Table 1. An example of dataset.

| Amaretto [Part] | Apple Juice [Part] | Cherry [Whole] | ... | White Cacao Liqueur [Part] | Tonic Water [Part] | Strawberry [Whole] | Strawberry [Piece] |
|--------------------|-----------------------|-------------------|-----|-------------------------------|-----------------------|-----------------------|-----------------------|
| 0.25 | 1.00 | 0.00 | ... | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.50 | 0.00 | ... | 0.00 | 0.00 | 0.00 | 4.00 |
| 1.00 | 0.00 | 1.00 | ... | 0.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | ... | 1.00 | 0.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | ... | 0.00 | 2.00 | 0.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | ... | 0.00 | 0.00 | 1.00 | 0.00 |
| 0.00 | 0.00 | 0.00 | ... | 0.00 | 0.00 | 0.00 | 0.00 |

3 Method

First of all, our system takes as inputs a collection of required ingredients, a collection of forbidden ingredients and whether “surprise” is necessary. Required ingredients are always used in recipes to be output finally, while forbidden ingredients are never used. “surprise” is like an indicator of a level of a concept close to the term serendipity used in the field of marketing. Regarding this part, we almost follow the work of Grace et al. [1,2]. However, there is only one point difference, which is that the input on “surprise” is not at that level and whether it is necessary or not.

On the other hand, our system consists of two processes: the generation of sets of ingredients and the generation of ingredient quantities (Fig. 1).

3.1 The Generation of Sets of Ingredients

In the generation of sets of ingredients, two deep learning models are used. One is a multilayer perceptron [3]. It takes as input a vector representation of a part of ingredients contained in a recipe and outputs levels of co-occurrence relation that each ingredient is included in the recipe. Another is a variational autoencoder [4] that encodes a vector with information such as which ingredient is included in a recipe into a vector of multidimensional normal distribution. First, the system receives a request in the form of a collection of required ingredients, a set of forbidden ingredients, and whether “surprise” is necessary. Next, using the multilayer perceptron, excluding forbidden ingredients, add an ingredient with the highest levels of co-occurrence relation of entering recipe or “surprise”

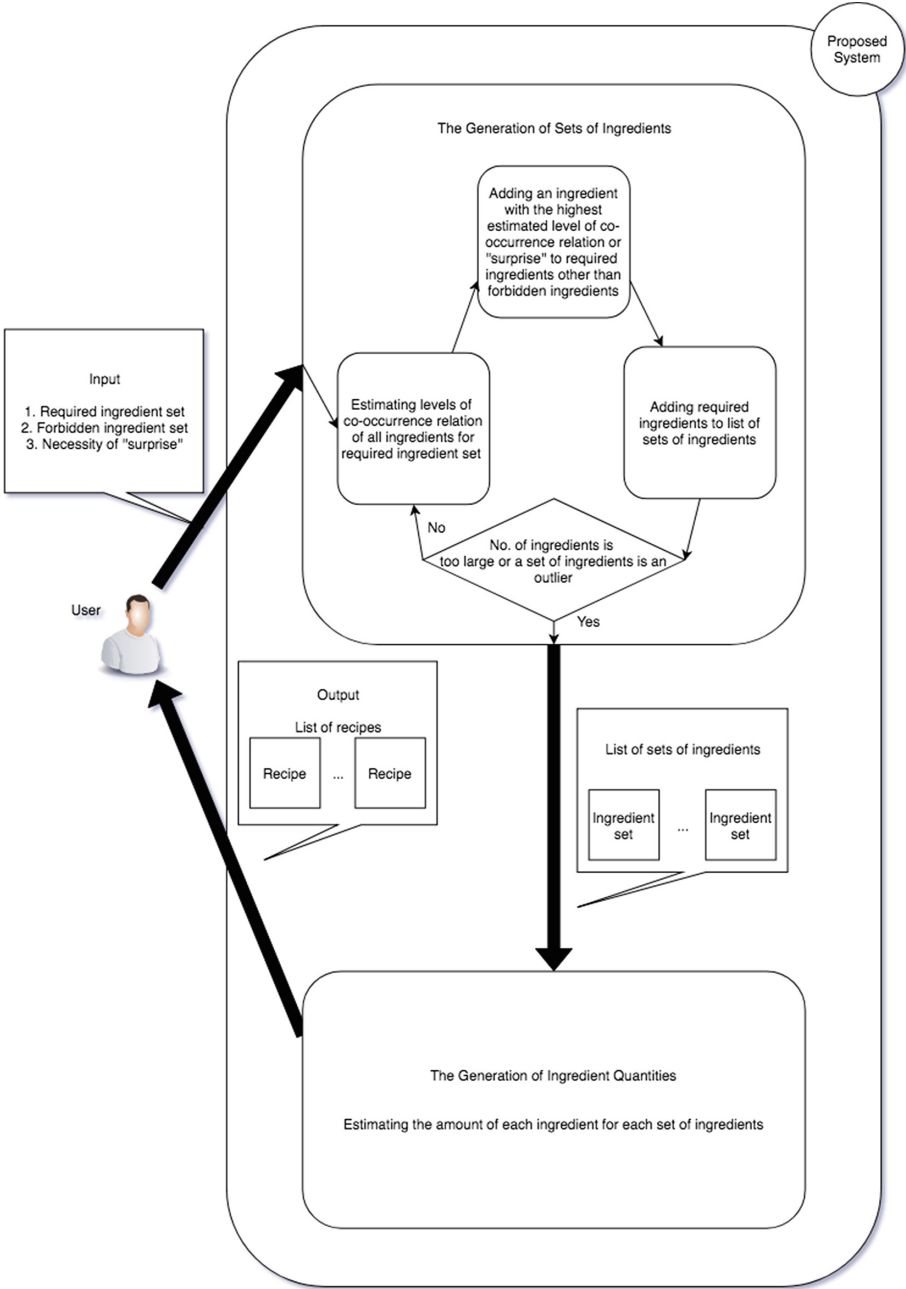


Fig. 1. Image of the proposed system.

estimate to the required ingredient set. The definition of “surprise” here [1,2] is as follows. When a set of other ingredients is given in multilayer perceptron and it estimates levels of co-occurrence relation P_θ of entering recipe of different ingredients, s expressed by the following equation is called “surprise”.

$$s(f|c) = -\log_2 \frac{P_\theta(f)}{P_\theta(f|c)} \quad (1)$$

where f is an ingredient and c is context of set of ingredients. Considering computational ease of use, the length of the combination at training is up to 3.

For a recipe to which a selected ingredient is added for an required ingredient set, outlier detection with the Mahalanobis distance and the chi-squared distribution [4] is performed on the vector coded by the variational autoencoder. If there is no outlier, a process such as addition of ingredient and processing of outlier detection is repeated. If it is judged as outlier or the number of ingredients exceeds the maximum value of the data set, this algorithm ends. We output a list including all the collections of ingredients from the required ingredient set to the final set of the repetition (however, those whose ingredient set size is 1 or 0 are removed from the candidates).

3.2 The Generation of Ingredient Quantities

The generation of ingredient quantities receives as input a list of ingredients outputted in the process of the generation of sets of ingredients. The following processing is performed on each set of ingredients. First, using the regression by multilayer perceptron [3], the system estimates amounts of each ingredient contained in the set of ingredients. Information on the estimate of quantity is added to the set of ingredients. A set of all ingredient sets subjected to the above processing is regarded as a final output result.

Loss Function in Learning of Multilayer Perceptron (MLP). When existing functions such as Mean Squared Error, Mean Absolute Error and MAPE are used as the loss function in learning of Multilayer Perceptron (MLP) for ingredient quantity generation, a problem that extremely small amounts (value close to 0) is output since the train data is sparse (amounts of ingredients that are not used is 0) arises. Therefore, in this study, we have defined the loss function L_N uniquely (2).

$$L_N = \frac{1}{NP} \sum_{n=1}^N \sum_{p=1}^P \frac{\text{sign}(y_p^{(n)}) |y_p^{(n)} - \hat{y}_p^{(n)}|}{\max(y_p^{(n)}, \epsilon)} \quad (2)$$

$$0 \leq y_p^n \leq 1, 0 \leq \hat{y}_p^{(n)} \leq 1, \epsilon > 0$$

where, N is the number of data, P is the dimension of data. ϵ is a tiny positive number to prevent division by 0, such as 10^{-7} is used. In MAPE, the loss for each sample is multiplied by the sign of the true value y_p^n . When y_p^n is 0, the loss is fixed to 0.

Evaluation Function in Validating of Multilayer Perceptron (MLP). If the proposed loss function 2 is used as an evaluation function in validating, there is a problem that it is easier to output a better evaluation value (lower value) as the number of ingredients used is smaller, that is, data with a lot of $y_p^{(n)}$ where $\text{sign}(y_p^{(n)})$ takes zero. Therefore, we also uniquely defined the evaluation function (3).

$$L_N = \frac{1}{\sum_{n=1}^N \sum_{p=1}^P \text{sign}(y_p^{(n)})} \sum_{n=1}^N \sum_{p=1}^P \frac{\text{sign}(y_p^{(n)}) |y_p^{(n)} - \hat{y}_p^{(n)}|}{\max(y_p^{(n)}, \epsilon)} \quad (3)$$

$$0 \leq y_p^n \leq 1, 0 \leq \hat{y}_p^{(n)} \leq 1, \epsilon > 0$$

By replacing NP with the sum of $\text{sign}(y_p^{(n)})$, it becomes equivalent to MAPE for $y_p^{(n)}$ which take positive values.

4 Experiment

Based on our method, we constructed a support system for recipe design on WEB application. Below are two screens of the application screen.

If you enter information on “required ingredients”, “forbidden ingredients” and “surprise”, you will transition to the result page (Fig. 2).

The input information is described at the top. A recommended recipe is displayed under that. Below it a plurality of recipes generated by the algorithm are displayed (Fig. 3).

First, We also generated recipes by our method using requests of randomly generated. After that, we took and analyzed a questionnaire including question items on the generated recipes.

In the questionnaire, the main questions are roughly divided into two types. One type is the question “Which of the following two recipes do you feel more than wanting to drink?” Another type is the question “Which of the following two recipes do you think is “a cocktail recipe created by a computer algorithm?””. Ten questions were prepared for each question type. The two recipes presented in one question are the recipe generated by the algorithm based on a certain request and the known recipe that most closely matches the same request. Furthermore, we asked questions such as taste for alcoholic beverages and drinking habits. A questionnaire survey was conducted for 37 males and females aged from 20 to 43. Using the data obtained from the questionnaire, we analyzed the probability of wanting to drink the recipe of the algorithm, the probability of discrimination between a recipe of the algorithm and a known recipe, respectively by logistic regression.

4.1 Procedure

Two patterns of analysis by logistic regression are carried out based on the questionnaire result. One is modeling the probability that the recipe of the algorithm is preferred, and another is modeling the probability that the recipe of the algorithm is distinguished.

Please enter your query for cocktail design:

Required ingredients:

Base spirits:

- Absolut Apeach
- Absolut Citron
- Absolut Kurant
- Absolut Mandrin
- Absolut Pears
- Absolut Raspberri
- Absolut Ruby Red
- Absolut Vanilla
- Absolut Vodka
- Aged Rum Of Cuban Type
- Apricot Brandy
- Bourbon
- Canadian Whiskey
- Cherry Brandy
- Cognac
- Dark Rum Of Jamaican Type
- Gin
- Light Rum
- Rye Whiskey
- Scotch Whiskey
- Tequila

Other spirits:

- Amaretto
- Banana Liqueur
- Benedictine Dom
- Bitters
- Blue Curacao
- Calvados
- Campari
- Cherry Liqueur
- Crème De Menthe, White
- Dark Cacao Liqueur
- Dry Vermouth
- Dubonnet
- Green Chartreuse
- Kahlúa
- Maraschino Liqueur
- Orange Bitters
- Orange Curacao
- Pastis
- Port, Red
- Sherry
- Sweet Vermouth
- Triple Sec
- Whisky/Honey Liqueur
- White Cacao Liqueur

Mixers:

- Apple Juice, Fresh Pressed
- Apple Juice
- Cranberry Juice
- Cream
- Ginger Ale
- Grapefruit Juice
- Grenadine
- Lemon Juice
- Lime Juice
- Orange Juice
- Passion Fruit Juice
- Pineapple Juice
- Simple Syrup
- Soda Water
- Tonic Water

Fruits & Berries:

- Cherry
- Lemon
- Lime
- Maraschino Cherry
- Orange
- Raspberry
- Strawberry

Other ingredients:

- Egg White
- Milk
- Mint Leaf
- Sugar, Superfine

Forbidden ingredients:

Base spirits:

- Absolut Apeach
- Absolut Citron
- Absolut Kurant
- Absolut Mandrin
- Absolut Pears
- Absolut Raspberri
- Absolut Ruby Red
- Absolut Vanilla
- Absolut Vodka
- Aged Rum Of Cuban Type
- Apricot Brandy
- Bourbon
- Canadian Whiskey
- Cherry Brandy
- Cognac
- Dark Rum Of Jamaican Type
- Gin
- Light Rum
- Rye Whiskey
- Scotch Whiskey
- Tequila

Other spirits:

- Amaretto
- Banana Liqueur
- Benedictine Dom
- Bitters
- Blue Curacao
- Calvados
- Campari
- Cherry Liqueur
- Crème De Menthe, White
- Dark Cacao Liqueur
- Dry Vermouth
- Dubonnet
- Green Chartreuse
- Kahlúa
- Maraschino Liqueur
- Orange Bitters
- Orange Curacao
- Pastis
- Port, Red
- Sherry
- Sweet Vermouth
- Triple Sec
- Whisky/Honey Liqueur
- White Cacao Liqueur

Mixers:

- Apple Juice, Fresh Pressed
- Apple Juice
- Cranberry Juice
- Cream
- Ginger Ale
- Grapefruit Juice
- Grenadine
- Lemon Juice
- Lime Juice
- Orange Juice
- Passion Fruit Juice
- Pineapple Juice
- Simple Syrup
- Soda Water
- Tonic Water

Fruits & Berries:

- Cherry
- Lemon
- Lime
- Maraschino Cherry
- Orange
- Raspberry
- Strawberry

Other ingredients:

- Egg White
- Milk
- Mint Leaf
- Sugar, Superfine

Is surprise necessary?:

Submit query

Fig. 2. The page for entering a query.

CocktailDesigner

Your query for cocktail design:

Required ingredients: Maraschino Cherry(Whole) / Maraschino Liqueur(Par)
 Forbidden ingredients: Ginger Ale(Par) / Soda Water(Par) / Egg White(Par) / Milk(Par) / Absolut Vodka(Par) / Dark Rum Of Jamaican Type(Par) / Gin(Par) / Light Rum(Par) / Tequila(Par)
 Surprise: Yes

[Submit another query](#)

Recipes:

Recommendation from known recipes:

Angel's Tit

- Cream(Par) 1.00
- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 1.00

Generated recipes:

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.67

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.67
- Green Chartreuse(Par) 0.67

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.50
- Green Chartreuse(Par) 0.67
- Pineapple(Wedge) 1

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.50
- Green Chartreuse(Par) 0.67
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.50
- Green Chartreuse(Par) 0.67
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.50
- Green Chartreuse(Par) 0.67
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.50

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.33
- Green Chartreuse(Par) 0.33
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.50
- Cherry Liqueur(Par) 0.50

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.33
- Green Chartreuse(Par) 0.33
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.50
- Cherry Liqueur(Par) 0.33
- Lime Juice(Par) 0.25

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.33
- Green Chartreuse(Par) 0.33
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.33
- Cherry Liqueur(Par) 0.33
- Lime Juice(Par) 0.25
- Aged Rum Of Cuban Type(Par) 1.00

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.25
- Green Chartreuse(Par) 0.25
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.33
- Cherry Liqueur(Par) 0.25
- Lime Juice(Par) 0.25
- Aged Rum Of Cuban Type(Par) 1.00
- Raspberry(Whole) 1.50

- Maraschino Cherry(Whole) 1.00
- Maraschino Liqueur(Par) 0.25
- Green Chartreuse(Par) 0.33
- Pineapple(Wedge) 1
- Cherry(Whole) 1.00
- Orange(Wedge) 1
- Pineapple Juice(Par) 0.33
- Cherry Liqueur(Par) 0.33
- Lime Juice(Par) 0.25
- Aged Rum Of Cuban Type(Par) 1.00
- Raspberry(Whole) 1.33
- Cognac(Par) 1.25

[Submit another query](#)

Fig. 3. The page displaying a result.

First, modeling the probability that the recipe of the algorithm is preferred is explained. The objective variable is set to 1 if the algorithm recipe is selected as the recipe that the user prefers, and 0 if not selected. We prepared the following variables as explanatory variable candidates.

- Type of alcohol drink preference (True: 1, False: 0)
 - The most popular alcohol drink (whisky or wine or beer, plum wine)
 - Popular alcohol drink (liqueur or jin or shochu or sake)
 - Unpopular alcohol drink (brandy or spirits or vodka or rum)
 - None
- Question
 - Question 1
 - Question 2
 - Question 3
 - Question 4
 - Question 5
 - Question 6
 - Question 7
 - Question 8
 - Question 9
 - Question 10
- Use of “surprise” in algorithm (True: 1, False: 0)
- Alcohol preference (True: 1, False: 0)
- Cocktail preference (True: 1, False: 0)
- Age (normalized)
- Sex (Male: 1, Female: 0)
- Alcohol drink preference (True: 1, False: 0)
 - Whisky
 - Jin
 - Beer
 - Liqueur
 - Wine
 - Sake
 - Plum wine
 - Shochu
- Experience equivalent to work of making cocktail (True: 1, False: 0)
 - Experienced
 - Not experienced
- Frequency of drinking cocktail
 - No drinking
 - Few (About 2 days or less per month)
 - Normal (About 1 to 3 days per week)
 - Frequent (About 4 days or more per week)
- Frequency of drinking alcohol
 - No drinking
 - Few (About 2 days or less per month)

- Normal (About 1 to 3 days per week)
- Frequent (About 4 days or more per week)
- Drinking amount on drinking alcohol
 - No drinking
 - Small (It is equivalent to beer medium bottle (1 bottle) or sake (1 go) or whisky single (2 cups) or less)
 - Normal (It is equivalent to beer medium bottle (1 to 2 bottles) or sake (1 to 2 go) or whisky single (3 cups))
 - Large (It is equivalent to beer medium bottle (3 bottles) or sake (3 go) or whisky double (3 cups)) or more
- Maximum value of “surprise” at the time of ingredient selection (normalized)

Variables are selected from the above variables using the variable increment method of the stepwise method based on AIC criterion and the resulting model is treated as an official model in this study.

We describe the modeling the probability that the recipe of the algorithm is distinguished. The objective variable is 1 if the recipe of the algorithm is selected as the recipe which is thought to have been created by the algorithm, and 0 if it is not chosen. For explanatory variables, we use the same variables as those used for modeling the probability that the recipe of the algorithm is preferred. However, note that the variables of the question correspond to different questions. In the same way as modeling the probability that the recipe of the algorithm is preferred, the variable selection is also done, and the model obtained from the result is handled as an official model in this study.

4.2 Results

Crosstabulation on whether or not each respondent like alcohol and whether or not each respondent like cocktails is shown below (Table 2).

Table 2. Crosstabulation on whether or not each respondent like alcohol and whether or not each respondent like cocktails.

| | No. of respondent who like cocktails | No. of respondent who do not like cocktails | Total |
|---|--------------------------------------|---|-------|
| No. of respondent who like alcohol | 21 | 6 | 27 |
| No. of respondent who do not like alcohol | 7 | 3 | 10 |
| Total | 28 | 9 | 37 |

Crosstabulation of preference of existing recipe vs. recipe created by algorithm and presence of “surprise” is shown below.

Table 3. Crosstabulation of preference of existing recipe vs. recipe created by algorithm and presence of “surprise.”

| | “surprise” | No “surprise” | Total |
|-----------------------------|------------|---------------|-------|
| Existing recipe | 66 | 115 | 181 |
| Recipe created by algorithm | 45 | 144 | 189 |
| Total | 111 | 259 | 370 |

As shown in Table 3, the recipe of the algorithm seems to prefer to drink is about 51%.

The result of modeling the probability that the recipe of the algorithm is preferred is shown in the following table (Table 4).

Table 4. The result of modeling the probability that the recipe of the algorithm is preferred.

| Coefficient | Estimate | z value | p value |
|--|----------|---------|---------|
| Intercept | -0.2644 | -1.210 | 0.22629 |
| Use of “surprise” in algorithm | 0.6280 | 2.682 | 0.00733 |
| Alcohol drink preference (liqueur) | 0.6640 | 2.356 | 0.01846 |
| Alcohol drink preference (beer) | -0.4654 | -2.099 | 0.03586 |
| Type of alcohol drink preference (unpopular alcohol drink) | -0.6522 | -1.825 | 0.06807 |

Crosstabulation of the recipe which was thought to be a recipe created by the algorithm and the presence of “surprise” is shown below (Table 5).

Table 5. Crosstabulation of the recipe which was thought to be a recipe created by the algorithm and the presence of “surprise.”

| | “surprise” | No “surprise” | Total |
|---------------------------------------|------------|---------------|-------|
| Recipe created by algorithm (Correct) | 113 | 47 | 160 |
| Existing recipe (Incorrect) | 146 | 64 | 110 |
| Total | 159 | 111 | 370 |

From the result, the rate at which the recipe of the algorithm is perceived is about 57%.

The result of modeling the probability that the recipe of the algorithm is distinguished is shown in the following table (Table 6).

Table 6. The result of modeling the probability that the recipe of the algorithm is distinguished.

| Coefficient | Estimate | z value | p value |
|--|----------|---------|---------|
| Intercept | 1.0116 | 2.450 | 0.01428 |
| Alcohol drink preference (whisky) | -0.7336 | -3.006 | 0.00265 |
| Alcohol drink preference (liqueur) | 0.8870 | 2.775 | 0.00553 |
| Cocktail preference | -1.1172 | -3.086 | 0.00203 |
| Frequency of drinking alcohol (few) | -0.6893 | -0.836 | 0.40330 |
| Frequency of drinking alcohol (normal) | -1.4085 | -1.786 | 0.07407 |
| Frequency of drinking alcohol (frequent) | -1.7200 | -2.027 | 0.04271 |
| Experience equivalent to work of making cocktail (experienced) | 1.1330 | 2.548 | 0.01082 |
| Type of alcohol drink preference (popular alcohol drink) | 1.4128 | 2.002 | 0.04531 |

5 Discussion

In this study, it is desirable that we proposed a system with the following properties. One is to make people want to drink cocktails of recipes generated by the algorithm. Another is that recipes generated by the algorithm is indistinguishable from existing recipes.

As an overall tendency, there are more cases where people want to drink a cocktail of recipes generated by the algorithm, and there are more cases that a recipe generated by the algorithm is distinguished from an existing recipe by a person. However, neither of these trends is prominent.

In order to make more appreciate recipe which is match to individual person's favorite, the following factors are important. First of all, it is better to use "surprise". In the system proposed in this study, it is left to the user whether or not to use "surprise", but we should always use "surprise" to reduce the elements that the user must select. By reducing the number of elements that must be selected, motivation for user selection and improvement in satisfaction after selection can be expected. Also, the features of users suitable for the proposed system are as follows.

- Person who likes liqueurs
- Person who does not like beer
- Person who does not like unpopular alcohol drink

People who can not distinguish between recipe generated by algorithm and existing recipe have the following features.

- Person who does not like liqueurs
- Person who likes cocktails

- Person who frequently drinks alcohol
- Person who has no experience equivalent to work making cocktails
- Person who likes popular alcohol drink

To summarize the discussions in the above two viewpoints, we point out following matters.

The likes and dislikes of liqueur in the features of users suitable for the proposed system are opposite, however, people who like liqueurs are thought to want to drink cocktails of algorithm recipes after distinguishing them from existing recipes. Therefore, liqueur lover is appropriate as a target user image. Even people who do not like popular alcohol drinks or unpopular alcohol drinks, some people prefer cocktails. Such people are suitable as users of our system. Such people are considered people who does not like drinking alcohol drink as it is. In other words, they are suitable people as a target for providing cocktails. People who do not like popular drinks and have no experience equivalent to a cocktail job match the user image. It is thought that such people are not familiar with alcohol drink or cocktail. Therefore, it should be assumed that the user does not have much knowledge. People with a high frequency of alcohol drinking are not only difficult to discriminate through recipes generated by algorithms or not, but also they should consume a large amount of alcohol. It can be said that these people can be promising customers from a commercial point of view.

6 Conclusion

We proposed a support system for demand driven design of cocktail recipes.

As an overall tendency, there were more cases where people want to drink a cocktail of recipes generated by the algorithm, and there were more cases that a recipe generated by the algorithm is distinguished from an existing recipe by a person. However, neither of these trends was prominent.

In addition, as a result of analyzing the questionnaire, the features of people ideal for users and the points to be improved of the system were also clarified.

At the present study, we adopted a system that allows users to select ingredients. However, in the present system, there are too many choices and it may become a burden to the user. Therefore, instead of letting users choose ingredients, we should allow users to enter rough information such as taste. Also, since we found “surprise” to be preferable, we will change to specifications that always use “surprise”.

Furthermore, we would like to conduct an evaluation experiment that people actually drink cocktails made based on recipes made by the proposed method.

Appendix: Questionnaire

In questions whose types are “Which of the following two recipes do you feel more than wanting to drink?” or “Which of the following two recipes do you think is “a cocktail recipe created by a computer algorithm”?”, a recipe generated by the

algorithm and an existing recipe are presented in a paired comparison format. Ten kinds of different recipe pairs were prepared for each of the two types of questions.

Next, questions of the questionnaire is shown with the answer format.

- Sex
 - Male
 - Female
- Age

An integer of 20 or more
- Do you like alcohol?
 - Yes
 - No
- Frequency of drinking
 - 7 days per week
 - 6 days per week
 - 5 days per week
 - 4 days per week
 - 3 days per week
 - 2 days per week
 - 1 day per week
 - 1 or 2 days per month
 - Many months I do not drink
 - I do not drink at all
- Drinking amount per day of drinking alcohol
 - It is equivalent to beer medium bottle (1 bottle) or sake (1 go (\doteq 200 ml)) or whisky single (2 cups) or less
 - It is equivalent to beer medium bottle (1 to 2 bottles) or sake (1 to 2 go) or whisky single (3 cups)
 - It is equivalent to beer medium bottle (3 bottles) or sake (3 go) or whisky double (3 cups)
 - It is equivalent to beer medium bottle (4 to 6 bottles) or sake (4 to 6 go) or whisky double (5 cups)
 - It is equivalent to beer medium bottle (7 to 10 bottles) or sake (7 go to 1 sho) or whisky bottle (1 bottle)
 - It is equivalent to beer medium bottle (over 10 bottles) or sake (over 1 sho) or whisky bottle (more than 1 bottle)
 - I do not drink at all
- What kind of alcohol drink you like
 - Whisky
 - Vodka
 - Gin
 - Spirits
 - Beer
 - Brandy
 - Lamb

- Liqueur
- Wine
- Sake
- Plum wine
- Shochu
- Other
- Do you like cocktails?
 - Yes
 - No
- Frequency of drinking cocktails
 - 7 days per week
 - 6 days per week
 - 5 days per week
 - 4 days per week
 - 3 days per week
 - 2 days per week
 - 1 day per week
 - 1 or 2 days per month
 - Many months I do not drink
 - I do not drink at all
- Do you have cocktail making work or equivalent experience?
 - Yes
 - No
- Which of the following two recipes do you feel more than wanting to drink?
 - Question 1
 - * Bourbon 1.50[Part]/Orange Juice 2.00[Part]
 - * Flowers (Edible) 1[Flower]/Bourbon 1.25[Part]/Blue Curacao 0.50[Part]
 - Question 2
 - * Bourbon, Peach Flavored 1.50[Part]/Raspberry Rum 0.50[Part]/Raspberry 2.67[Whole]
 - * Bourbon, Peach Flavored 2.00[Part]/Lemon 1[Twist]
 - Question 3
 - * Brandy 1.00[Part]/Galliano 1.00[Part]
 - * Williams Pear Liqueur 0.33[Part]/Brandy 1.00[Part]/Almond Liqueur 0.50[Part]
 - Question 4
 - * Gin 1.50[Part]/Ginger Beer 5.00[Part]
 - * Tamarind Juice 0.50[Part]/Milk, Frozen In Cubes 0.67[Part]/Mint Leaf 1[Sprig]
 - Question 5
 - * Maraschino Cherry 1.00[Whole]/Agave Nectar 0.33[Part]/Lime Juice 0.67[Part]
 - * Absolut Vodka 2.00[Part]/Campari 1[Dash]/Maraschino Cherry 1.00[Whole]

- Question 6
 - * Apple Juice 1.00[Part]/Butter 1.00[Teaspoon]/Dark Rum (Aged) 1.00[Part]/Lemon 1.00[Slice]
 - * Absolut Vodka 1.50[Part]/Apple Juice 5.00[Part]/Blue Curacao 0.75[Part]/Lemon 1[Wheel]
- Question 7
 - * Absolut Peppar 1.50[Part]/Tomato 1.00[Slice]
 - * Pink Peppers 10.75[Whole]/Tomato 1[Quarter]/Olive Juice 0.25[Part]
- Question 8
 - * Cherry 1.00[Whole]/Pineapple Liqueur 0.25[Part]/Cream Of Coconut 0.50[Part]
 - * Cherry 1.00[Whole]/Light Rum 2.00[Part]/Soda Water 4.00[Part]
- Question 9
 - * Absolut Vanilia 0.75[Part]/Calvados 0.75[Part]/Cinnamon Syrup 2[Dash]
 - * Lemon Popsicle 0.75[Whole]/Tea 1.50[Part]/Simple Syrup 0.33[Part]
- Question 10
 - * Star Anise 1.00[Slice]/Chocolate 1.00[Whole]/Simple Syrup 0.33[Part]/Light Rum 1.00[Part]
 - * Chocolate 1.00[Whole]/Dark Cacao Liqueur 0.33[Part]/Port, Red 1.00[Part]/Yellow Chartreuse 0.33[Part]
- Which of the following two recipes do you think is “a cocktail recipe created by a computer algorithm”?
 - Question 1
 - * Milk 4.00[Part]/Pastis 1.00[Part]
 - * Pastis 1[Dash]/Bitter Lemon 0.75[Part]/Lemon 1[Twist]
 - Question 2
 - * Elderflower Liqueur 0.50[Part]/Tomato Juice 1.25[Part]/Absolut Cilantro 1.25[Part]
 - * Absolut 100 1.50[Part]/Lemon Juice 2[Dash]/Tomato Juice 5.00[Part]
 - Question 3
 - * Bourbon 1.00[Part]/Maraschino Liqueur 3[Dash]/Triple Sec 0.50[Part]
 - * Falernum 0.25[Part]/Watermelon Juice 1.33[Part]/Melon 1.00[Slice]
 - Question 4
 - * Dry Vermouth 2.00[Part]/Pastis 2[Dash]
 - * Coffee Beans 3.00[Whole]/Pastis 1[Dash]/Dark Rum Of Jamaican Type 0.75[Part]
 - Question 5
 - * Absolut Apeach 1.50[Part]/Lemon-Lime Soda 5.00[Part]/Lime 1[Wedge]
 - * Banana 1.00[Slice]/Absolut Apeach 1.00[Part]/Cream 0.67[Part]
 - Question 6
 - * Bourbon 1.00[Part]/Cognac 1.00[Part]/Milk, Frozen In Cubes 4.00[Part]/Milk 4.00[Part]

- * Milk, Frozen In Cubes 0.75[Part]/Amontillado Sherry 0.25[Part]/Bourbon 0.50[Part]/Cognac 0.50[Part]
- Question 7
 - * Orange Juice 0.67[Part]/Apricot 1.25[Whole]/Dry Vermouth 0.33[Part]
 - * Champagne 3.00[Part]/Orange Juice 2.00[Part]
- Question 8
 - * Saffron Sugar Syrup 0.25[Part]/Canadian Whisky 1.33[Part]/Orange Juice 1[Dash]
 - * Canadian Whisky 1.00[Part]/Cordial Medoc 1.00[Part]/Dry Vermouth 1.00[Part]
- Question 9
 - * Absolut Vodka 0.33[Part]/Almond Liqueur 0.50[Part]/White Cacao Liqueur 0.50[Part]
 - * Green Apple Liqueur 0.50[Part]/Garlic Salt [To Taste]/Balsamico Vinegar 0.25[Part]
- Question 10
 - * Absolut Vodka 1.50[Part]/Cucumber 1.00[Slice]/Sake 0.50[Part]
 - * Cinnamon Cane 1.00[Whole]/Vanilla Pod 1.00[Whole]/Honey 0.33[Part]

In Question 2, Question 5, Question 6, Question 8, and Question 10, whose type is “Which of the following two recipes do you feel more than wanting to drink?” and Question 2, Question 7 and Question 8, whose type is “Which of the following two recipes do you think is “a cocktail recipe created by a computer algorithm”?”, the recipes created by the proposed algorithm is arranged above, other than the recipe created by the algorithm is placed below. Also, in Question 2, Question 3, Question 5, Question 6, Question 7, Question 9 and Question 10, whose type is “Which of the following two recipes do you feel more than wanting to drink?” and Question 5, Question 7 and Question 10, whose type is “Which of the following two recipes do you think is “a cocktail recipe created by a computer algorithm”?”, “surprise” is used for the recipes created by the algorithm.

Second, we describe a method of preparing a pair of recipes. First, a virtual user’s request is generated by random number. For this request, a pair of a recipe generated by an algorithm and an existing recipe is prepared. For the recipe generated by the algorithm, the user’s request is given as input of the proposed system. However, since multiple recipes are generated in the proposed system, one recipe selected by random number from recipes of an output of the system is set as a recipe generated by the algorithm. For existing recipes, we chose a recipe that is most suitable for an user’s request by the method described later from the existing recipes we have.

Third, the generation of user’s requests is described. Regarding ingredients, the number of elections is decided for each required or forbidden ingredient by random number, and required ingredients and forbidden ingredients are selected as many as the determined number. For “surprise” as well, we decide whether or not to use it by random number.

Lastly, the selection of an existing recipe is described. The selection of an existing recipe uses only the information related to the ingredients of the user's request. Specifically, it selects based on values of an index d that is dissimilarity to the user's request based on the distance from the required ingredients entered, plus the penalty based on the set of forbidden ingredients. We defined the index by the following equation.

$$d(A, B, C) = 1 - \text{Dice}(A, B) + |A \cap C| \quad (4)$$

where, A is a set of ingredients used for one existing recipe, B is a set of required ingredients in request, C is a set of forbidden ingredients in request, $\text{Dice}(\cdot, \cdot)$ is Dice coefficient.

References

1. Grace, K., Maher, M.L., Wilson, D.C., Najjar, N.A.: Combining CBR and deep learning to generate surprising recipe designs. In: Goel, A., Díaz-Agudo, M.B., Roth-Berghofer, T. (eds.) ICCBR 2016. LNCS (LNAI), vol. 9969, pp. 154–169. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-47096-2_11
2. Grace, K., Maher, M.L.: Surprise-triggered reformulation of design goals. In: AAAI, pp. 3726–3732 (2016)
3. LeCun, Y.A., Bottou, L., Orr, G.B., Müller, K.-R.: Efficient BackProp. In: Montavon, G., Orr, G.B., Müller, K.-R. (eds.) Neural Networks: Tricks of the Trade. LNCS, vol. 7700, pp. 9–48. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-35289-8_3
4. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint [arXiv:1312.6114](https://arxiv.org/abs/1312.6114), 14 p. (2013)
5. Absolut Drinks: Discover the best cocktails and drink recipes. <https://www.absolutdrinks.com/en/>. Accessed 23 Dec 2018
6. Iyengar, S., Lepper, M.: When choice is demotivating: can one desire too much of a good thing? *J. Pers. Soc. Psychol.* **79**, 995–1006 (2000)
7. Filzmoser, P.: A multivariate outlier detection method. In: Aivazian, S., Filzmoser, P., Kharin, Y. (eds.) Proceedings of the Seventh International Conference on Computer Data Analysis and Modeling, vol. 1, pp. 18–22. Belarusian State University, Minsk (2004)