Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Quantifying Recipe Difficulty Based on Recipe Features

Sae Okatani
Nara Women's University
Naoko Yoshii
Nara Women's University
Masami Takata
takata@ics.nara-wu.ac.jp

Nara Women's University

Research Article

Keywords: Recipe Difficulty, Correlation Analysis, Regression Analysis, Rank–Distance, Questionnaire Survey

Posted Date: November 12th, 2026

DOI: https://doi.org/10.21203/rs.3.rs-11506878/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Additional Declarations: No competing interests reported.

Quantifying Recipe Difficulty Based on Recipe Features

Sae Okatani[1*], Naoko Yoshii[1] and Masami Takata[2*]

[1*]Faculty of Human Life and Environment, Nara Women's University, Kitauoya–Nishi–machi, Nara, 6308506, Nara, Japan.
[2*]Research Group of Information and Communication Technology for Life, Nara Women's University, Kitauoya–Nishi–machi, Nara, 6308506, Nara, Japan.

*Corresponding author(s). E-mail(s): takata@ics.nara-wu.ac.jp;
Contributing authors: nao4ji20@gmail.com;

Abstract

Recipe–sharing sites have become widespread, and platforms such as Cookpad host millions of recipes; however, recipe difficulty is not specified. Therefore, it is difficult to select recipes that match their skill level. In this paper, we propose a computation method for quantifying recipe difficulty based on recipe details. The method comprehensively considers six features: cooking action level, the number of characters in the recipe, the number of cooking utensils, number of seasonings, number of ingredients, and number of steps. Each feature is normalized, and mutually correlated features are integrated using regression coefficients to avoid double–counting overlapping information. The validity of the proposed method is confirmed through a questionnaire survey with **30** respondents and verified by the Friedman test and four rank–distance metrics. The proposed method produces a strict total order over recipes, a property required for fine-grained recipe recommendation but not provided by single–feature baselines.

Keywords: Recipe Difficulty, Correlation Analysis, Regression Analysis, Rank–Distance, Questionnaire Survey

1 Introduction

With the widespread use of the Internet, recipe–sharing websites have become popular. Cookpad [1], the largest recipe–sharing website in Japan, hosts a vast number of recipes and allows users to search for recipes from an extensive selection; approximately 1.7 million of the recipes have been released as a research dataset [2]. However, as the number of recipes continues to grow, it is not easy to select recipes suited to their skill level.

Cooking at home is closely related to dietary quality. Frequently cooking dinner at home is associated with a healthier diet [3], and cooking skills correlate positively with vegetable consumption and negatively with convenience–food consumption [4]. Food preparation is also one of the four domains of food literacy, together with planning and management, selection, and eating [5]. At the same time, time pressure, the desire for effortless meals, and the experience of "kitchen disasters" are reported as barriers to cooking from basic ingredients, whereas greater self–efficacy in one's cooking ability acts as a facilitator [6]. These findings suggest that cooking skills differ among individuals and that presenting recipes whose difficulty matches the skill of each user is important for supporting home cooking.

Computational support for food choice has been studied actively in the field of food computing [7], and food recommender systems are one of its central topics [8, 9]. Existing systems recommend recipes mainly based on user preferences estimated from ratings [10] and relationships among ingredients [11], and the healthiness of the recipes recommended by such systems has also been analyzed [12]. In conventional recipe search systems, ingredients or cooking time are likewise commonly used as search criteria [10, 11]. However, whether the user can actually prepare the presented recipe has received comparatively little attention, and search functions based on recipe difficulty are insufficient. Therefore, users often select recipes that require complex cooking techniques, which can lead to failed attempts or giving up midway through the cooking process.

To take difficulty into account in recipe search and recommendation, it must first be quantified from the recipe itself. Recipe texts have become an established target of natural language processing, and methods for converting a recipe into a structured representation, such as a flow graph [13] or an action graph [14], have been proposed. These studies show that cooking actions, ingredients, and tools can be extracted automatically from recipe text. However, their goal is to understand the structure of cooking procedures, and they do not evaluate how difficult the procedures are.

Several studies have addressed recipe difficulty directly. References [15–17] have proposed recipe difficulty based solely on cooking actions. In [15–17], difficulty is determined from cooking actions classified into four levels, and most recipes are concentrated at the lower difficulty levels. Consequently, users cannot effectively compare recipe difficulty. Other studies combine several recipe details, such as the numbers of ingredients, seasonings, and steps [18–20]; however, the details are simply added, or the recipes are grouped into a few levels, without examining the overlap of information among the details. Few studies have comprehensively considered other recipe–related details and integrated them into a fine–grained difficulty value that can be computed automatically for a large recipe collection .

In this paper, to quantify recipe difficulty, we propose a method that considers not only cooking actions but also the number of characters, cooking utensils, seasonings, ingredients, and steps. By normalizing and integrating these details, we achieve a recipe difficulty that more accurately reflects practical cooking conditions. The contributions of this paper are as follows:

- Six features are extracted automatically from recipe text and computed for approximately 605,000 Cookpad recipes, so that difficulty is obtained as a continuous value rather than as a few discrete levels.
- The correlations among the features are analyzed, and correlated features are integrated using regression slopes, which avoids double–counting the information shared among them.
- The computed difficulty is compared with the difficulty perceived by 30 questionnaire respondents using four rank–distance metrics, and the proposed method is shown to produce a strict total order over recipes, which single–feature baselines do not provide.

The remainder of this paper is organized as follows. Section 2 reviews previous research and clarifies the position of this study. Section 3 describes the proposed method. Section 4 presents the correlation and regression analyses and the resulting difficulty formulas. Section 5 reports the experiments, and Section 6 concludes the paper.

2 Previous Research

This section reviews four research areas related to this study: food and recipe recommender systems, cooking skills and food literacy, natural language processing of recipe texts, and the quantification of recipe difficulty. The position of this study is then summarized.

2.1 Food and Recipe Recommender Systems

Min et al. [7] survey food computing, which covers the perception, recognition, retrieval, recommendation, and monitoring of food using heterogeneous food data. Within this field, Tran et al. [8] give an overview of recommender systems in the healthy food domain and classify them into four types: those considering user preferences, those considering nutritional needs, those balancing preferences and nutritional needs, and those for groups of users. Bondevik et al. [9] systematically review 67 studies on food recommender systems and report that most systems employ content–based filtering and machine learning.

As individual methods, Freyne and Berkovsky [10] generate personalized recipe recommendations by relating user ratings of recipes to the ingredients contained in them. Teng et al. [11] construct ingredient networks from online recipes and show that these networks are effective for predicting recipe ratings. Trattner and Elsweiler [12] analyze recipes from a large recipe website against the nutritional standards of the World Health Organization and the United Kingdom Food Standards Agency and show that commonly used recommender algorithms tend to promote unhealthy recipes.

In these studies, the suitability of a recipe for a user is judged mainly from the viewpoints of preference and health. The effort and skill required to prepare the recipe are not modeled explicitly, and a fine–grained difficulty measure that could be supplied to such systems is not available.

2.2 Cooking Skills and Food Literacy

Vidgen and Gallegos [5] define food literacy through a Delphi study with food experts and interviews with young people, and identify eleven components organized into four domains: planning and management, selection, preparation, and eating. Hartmann et al. [4] develop a cooking skills scale and, using data from 4,436 participants in Switzerland, show that cooking skills correlate positively with weekly vegetable consumption and negatively with convenience–food consumption. Wolfson and Bleich [3] analyze data from 9,569 adults in the United States and show that cooking dinner frequently at home is associated with lower consumption of energy, fat, and sugar. Lavelle et al. [6] interview 27 adults and report barriers to cooking from basic ingredients, such as time pressure, the desire for effortless meals, and the effect of kitchen disasters, together with facilitators, such as greater self–efficacy in one's cooking ability.

These studies measure and characterize the skills of people who cook. To match recipes to people, a complementary measure of the difficulty of each recipe is required. This study addresses the recipe side of this matching.

2.3 Natural Language Processing of Recipe Texts

Recipes are a representative example of procedural text and have been studied in natural language processing. Harashima et al. [2] construct a dataset of approximately 1.7 million recipes and 36,000 meals from Cookpad and release it as an infrastructure for food research; this dataset is used in this study. Mori et al. [13] build a corpus in which 266 Japanese recipes are manually annotated as flow graphs, i.e., directed acyclic graphs whose vertices are foods, tools, and actions. Kiddon et al. [14] propose an unsupervised method that maps a recipe to an action graph, which defines what actions are performed on which objects and in what order.

These studies aim to interpret the structure of cooking procedures, and the difficulty of the procedures is outside their scope. They nevertheless show that cooking actions, ingredients, and tools can be identified in recipe text. In this study, the features required for difficulty are levels and counts; therefore, morphological analysis and dictionary matching are sufficient, which makes the method applicable to hundreds of thousands of recipes without manual annotation.

2.4 Quantification of Recipe Difficulty

Several previous studies examine the difficulty of cooking recipes.

In [15], Makino et al. propose a method for computing recipe difficulty for recipe search based on cooking actions. To obtain the information underlying this method, they extract 485 types of cooking actions from home economics textbooks approved by the Ministry of Education, Culture, Sports, Science and Technology (for elementary, junior high, and high schools, as well as vocational programs) and classify each

into one of four difficulty levels (1–4). They propose nine methods for computing recipe difficulty based on the cooking actions derived from the recipes. The methods quantify difficulty using such the mode, median, maximum value, mean, weighted difficulty level, frequency of cooking actions, and specificity of cooking actions. Evaluation experiments confirm that the method using the maximum value outperforms the others. However, a questionnaire survey reveals that respondents believe "few steps", "few ingredients", and "short preparation time" influence difficulty, suggesting that cooking actions alone are insufficient for adequately evaluating difficulty. Kusu et al. [16], from the same research group, report this method in English and evaluate it through a questionnaire survey of 140 respondents on a crowdsourcing site; the respondents can distinguish recipe lists ordered by the calculated difficulty from randomly ordered lists. The reasons given by the respondents for feeling that a recipe is difficult indicate that difficulty is determined not only by cooking activities but also by the cooking time and by the numbers of words, photographs, and ingredients in a recipe.

In [17], Iwamoto et al. propose a search system that computes the difficulty level of a recipe based on the number of cooking actions, taking into account the difficulty of each action. This method is based on the four levels (Level 4 to Level 1) of the Home Cooking Skills Certification [21]. The recipe difficulty score is computed by combining the frequency of occurrence and the weight of cooking actions for each difficulty level. In an evaluation experiment, nine participants are asked to compare a system that takes difficulty into account with one that does not. As a result, six participants support the system that takes difficulty into account, while three respond that they cannot decide. This suggests that difficulty–based rankings may be useful.

Other studies use recipe details in addition to cooking actions. Yajima and Kobayashi [18] propose a system that recommends "easy" recipes according to the conditions of the user, such as the ingredients and seasonings on hand. They score each recipe by adding normalized points for its seasonings, ingredients, and cooking verbs. Müller and Bergmann [19] define a complexity measure for recipes represented as workflows in a case–based reasoning system that generates recipes. The measure adds five normalized criteria: the number of ingredients, the number of preparation steps, the complexity of ingredient processing, the difficulty of the preparation steps, and the preparation duration. The difficulty and duration of each preparation step are assigned manually, and the measure is applied to 61 manually modeled sandwich recipes. Takahashi et al. [20] estimate cooking difficulty for recipe search by clustering 100 recipes using four indicators: cooking time, the number of steps, the number of ingredients, and the number of cooking actions. The recipes are classified into three levels (easy, normal, and difficult), and a survey of 18 respondents shows that the estimated level agrees with the perceived level in approximately 60% of cases.

2.5 Position of This Study

The action–based methods [15–17] share the common approach of evaluating recipe difficulty by focusing on cooking actions. However, they consider only cooking actions and do not sufficiently account for other details such as the number of steps, number of ingredients, and time required. The multi–feature methods [18–20] consider such details, but they add the features with equal weight or group the recipes into a few

levels. As shown in Section 4, features extracted from the same recipe are correlated; for example, the number of characters and the number of steps largely share the same information, so a simple sum overcounts it. In addition, the method in [19] depends on manually modeled workflows and manually assigned step difficulties, and the method in [20] assigns one of three levels, so recipes within the same level cannot be compared.

In contrast, the method proposed in this paper (i) extracts six features automatically from recipe text, (ii) integrates correlated features using regression slopes so that shared information is not double–counted, (iii) outputs difficulty as a continuous value that orders recipes without ties, and (iv) is computed for approximately $605,000$ recipes and compared with the difficulty perceived by questionnaire respondents. With respect to the research areas reviewed above, this study provides the recipe–side difficulty measure that is not available in food recommender systems (Section 2.1) and that complements the person–side measures of cooking skills (Section 2.2), using text processing of recipes (Section 2.3).

3 Proposed Method

This section describes the proposed method for quantifying recipe difficulty as a fine–grained continuous value. Representing difficulty as a continuous value allows users to more easily compare multiple recipes by dividing recipe difficulty into arbitrary levels.

3.1 Features Extracted from Recipes

Two approaches can be considered for computing difficulty: a bottom–up and a top–down approach. The computation formula is expressed as an equation based on features extracted from recipes.

In the bottom–up method, the relationship between a pre–ranked set of recipes and relevant features is examined. Specifically, multiple regression analysis is performed using the rank as the dependent variable and quantified feature information as explanatory variables. In this case, pre–ranked recipes are required; however, as discussed in [15, 17], existing rankings are not subdivided into clearly distinct levels and exhibit significant bias, making them unsuitable as a dependent variable. It is also possible to conduct a survey to establish rankings and perform multiple regression analysis based on those rankings, but ensuring sufficient accuracy would require a very large number of surveys.

In the top–down method, the formula is constructed using only the explanatory variables. The difficulty level can be computed by determining appropriate coefficients for each explanatory variable based on the statistical relationships among them.

In this study, the formula is constructed based on the top–down method, the following procedure is followed:

1. Extraction of details
2. Quantification of features
3. Normalization of each feature
4. Verification of correlations between quantifications
5. Integration using the slope of the regression line when correlation is confirmed
6. Construction of a formula using the integrated variables

The following six features are extracted from each recipe:
(a) Cooking action level
(b) Number of characters in the recipe
(c) Number of cooking utensils
(d) Number of seasonings
(e) Number of ingredients
(f) Number of steps

3.2 Extraction and Quantification of Features

3.2.1 Creating a Dictionary

To extract action levels, a dictionary of cooking actions and their difficulty levels is required. In this study, frequency analysis of approximately 1.7 million Cookpad entries is used to construct a cooking–action dictionary. The dataset includes recipe details, some of which are missing. These missing details do not affect the vocabulary in the cooking–procedure text. To extract cooking actions from recipe instructions, morphological analysis is conducted. Because the recipes on Cookpad are written in Japanese, cooking actions are extracted in two patterns: "*sa*"–variant nouns combined with the verb "*suru*" (e.g., "*chohri–suru*", meaning "to cook") and standalone verbs. Light verbs (such as "*suru*" used alone), non–cooking verbs related to general eating behavior (such as "*taberu*", meaning "to eat"), and intransitive verbs that do not denote cooking actions (such as "*tokeru*", meaning "to melt" in its intransitive sense) are excluded.

Approximately 3,150 distinct verbs are extracted. Frequency analysis shows that the top 100 words account for the vast majority of occurrences. Therefore, the dictionary is constructed using the following procedure:

- Prioritized selection of high-frequency words from the top occurrences.
- Normalization of spelling variations (definition of a canonical form).
- Exclusion of non-cooking verbs.

Table 1 Difficulty levels of cooking actions.

Level	Words	Ratio [%]	Examples
1	48	40.7	Add, Chop, Mix
2	58	49.2	Stir–fry, Pan–fry, Simmer
3	8	6.8	Deep–fry, Strain
4	4	3.4	Chamfer
Total	118	100	Coverage rate: 78.44%

The resulting dictionary comprises 118 words (113 distinct canonical forms after normalization), covering 78.44% of all cooking actions in the dataset. Table 1 shows the number of cooking actions included in the dictionary, their percentages by difficulty

level, and examples of each action. As shown in Table 1, the difficulty of each cooking action is defined on a four–level scale (1: Basic, 2: Standard, 3: Advanced, 4: Specialized) based on the Home Cooking Skills Certification of Japan [21].

To extract seasonings and cooking utensils, a dictionary–matching method is used. Unlike cooking actions, the types of seasonings and ingredients are relatively limited; therefore, frequency analysis is not performed. Instead, a list of seasonings commonly used in general cooking is created in advance. The seasoning list includes 261 types, including basic seasonings (salt, sugar, soy sauce, etc.), spices, and cooking oils, accounting for spelling variations. The cooking utensil list includes 139 types of cooking utensils, including "frying pan", "microwave", and "kitchen knife", also accounting for spelling variations.

3.2.2 Extraction of Features

To determine the difficulty level based on cooking–actions, morphological analysis is performed using the dictionary created.

This allows cooking–actions to be extracted from recipe descriptions. For each recipe, the cooking–action level (a) is defined as the maximum difficulty level among the cooking actions extracted from its instructions, following [15], where the maximum–based method outperforms other aggregation methods. Cooking actions not registered in the dictionary are excluded from this computation. To determine the character count, the total number of characters in the recipe instructions is counted. Spaces and line breaks are excluded; only actual characters are counted. Recipes with a high character count suggest that the instructions are detailed and the steps are complex. Generally, recipes with a higher character count contain more information necessary for cooking and may take longer to understand.

In the cooking utensil extraction process, a string search is performed on the text describing the cooking steps in a recipe to find terms registered in the cooking utensil dictionary. The extracted nouns are cross–referenced with the list, and the number of matching types is counted. Even if the same cooking utensil appears multiple times within a single recipe, it is counted as only one. Finally, the number of different types of cooking utensils used in each recipe is determined. It is believed that the greater the number of different types of cooking utensils, the more complicated the preparation becomes and the greater the difficulty of the recipe. For example, comparing a recipe that can be completed with "just a knife and a cutting board" to one that requires "a frying pan, a pot, an oven, and a mixer", the latter can be judged to be more difficult because it involves more effort in both preparation and cleanup.

The method for identifying seasonings is the same as that used for cooking utensils: a string search is performed on the recipe's ingredient list and cooking instructions; terms that match the seasoning dictionary are detected, and the number of different types is counted. It is believed that the greater the number of different seasonings, the more complex the seasoning becomes and the greater the difficulty of the recipe. Compared to recipes with simple flavors like salt or soy sauce, recipes that combine multiple seasonings to adjust the flavor are more challenging. Since typical recipes include an ingredient list, the number of items is counted directly from this list. However, since seasonings such as "salt", "pepper", and "soy sauce" may also appear in the

ingredient list, those classified as seasonings are excluded from the ingredient count and counted separately as seasonings. This allows for a clear distinction between seasonings and ingredients. The greater the number of different types of ingredients, the more time–consuming shopping and prep work become, suggesting that the cooking difficulty increases. Furthermore, recipes with many ingredients involve more details to manage — such as how to cut the ingredients and the timing of cooking — which can make them feel complex.

The number of steps is defined as the number of cooking steps in a recipe. A recipe with a high number of steps indicates that the process is complex and that cooking takes more time. Furthermore, the more steps there are, the more important it is to perform each step correctly, and the higher the likelihood of mistakes, so the recipe is considered more difficult.

3.2.3 Normalization Method

The six extracted features have different scales. For example, while the cooking–action–based difficulty ranges from 1 to 4, the number of characters ranges from several hundred to several thousand. To integrate these features and compute the overall difficulty; all features must be normalized to the same scale.

In this study, Min–Max normalization is adopted. It converts each feature's value to the range 0 to 1, as expressed by the following equation.

$$\text{Normalized value} = \frac{\text{Original value} - \text{Minimum value}}{\text{Maximum value} - \text{Minimum value}}.$$

This normalization ensures that all features take values within the range 0 to 1, enabling comparison and integration among the features. The closer the normalized value is to 0, the easier the task is for that feature; the closer it is to 1, the more difficult it is.

4 Correlation Analysis and Regression Analysis

In this section, correlation and regression analyses are performed to identify relationships among the six features and to derive an appropriate integration method. This study uses data obtained from Cookpad, the largest recipe–sharing website in Japan. The data are acquired from a research dataset [1] provided by the National Institute of Informatics. The dataset contains basic recipe information, including titles, descriptions, ingredient lists, and cooking instructions. Although the dataset contains approximately 1.7 million recipes, each feature is computed for approximately $605,000$ recipes that contain all required information without missing values. As stated in Section 3.2, the entire 1.7–million–entry dataset is used only for constructing the cooking–action dictionary, where vocabulary coverage rather than metadata completeness is the relevant criterion.

To extract cooking actions, the morphological analysis tool `Fugashi` [22] is used. `Fugashi` enables high–speed morphological analysis of Japanese text and is built upon `MeCab` [23].

4.1 Correlation Analysis

To clarify the relationships among the features, Pearson correlation is computed for the six normalized features. Table 2 shows the resulting correlation matrix. Each column

Table 2 Correlation coefficients among the six features.

	(a)	(b)	(c)	(d)	(e)	(f)
(a)	1.00	0.13	0.10	0.09	0.12	0.13
(b)	0.13	1.00	0.58	0.25	0.36	0.86
(c)	0.10	0.58	1.00	0.14	0.18	0.49
(d)	0.09	0.25	0.14	1.00	0.29	0.20
(e)	0.12	0.36	0.18	0.29	1.00	0.32
(f)	0.13	0.86	0.49	0.20	0.32	1.00

represents the following: (a) the cooking–action level, (b) the number of characters, (c) the number of cooking utensils, (d) the number of seasonings, (e) the number of ingredients, and (f) the number of steps.

The analysis shows that the cooking–action level (a) has only a very weak correlation with the other features ($0.09 \leq |r| \leq 0.13$). This indicates that the cooking–action level is essentially independent of the other features: recipes with a high cooking–action level do not necessarily have many steps, and conversely, recipes with many steps do not necessarily include difficult actions.

In contrast, the number of characters (b) shows moderate–to–strong correlations with the number of steps (f) ($r = 0.86$), the number of cooking utensils (c) ($r = 0.58$), and the number of ingredients (e) ($r = 0.36$). The high correlation between (b) and (f) in particular indicates that information about "having many steps" is largely shared between the two features. Therefore, simply summing all six normalized features would overcount step–related information. An integration method that explicitly accounts for these correlations is necessary.

4.2 Regression Analysis

To quantify the redundancy between (b) and the correlated features, simple linear regression is performed with the number of characters as the independent variable and each correlated feature as the dependent variable. This regression quantifies how much the character count explains each feature and yields the slope coefficients required for the subsequent integration.

Table 3 shows the resulting regression equations. Here, x denotes the normalized character count, y denotes the normalized value of each correlated feature, and R^2 is the coefficient of determination.

4.3 Quantifying Recipe Difficulty

Using the regression results, intermediate variables that integrate the correlated features without double–counting their shared information are defined. The general

Table 3 Regression results between the number of characters and each correlated feature.

Feature	Correlation coefficient	R^2	Regression equation
(c)	0.581	0.337	$y = 1.9384x + 0.0305$
(d)	0.245	0.060	$y = 0.9102x + 0.1489$
(e)	0.362	0.131	$y = 0.8263x + 0.0916$
(f)	0.857	0.735	$y = 0.7276x + 0.0046$

principle is as follows. When two normalized features x and y are correlated and the regression line of y on x has slope α, then αx represents the portion of y explained linearly by x. A weighted average of αx and y therefore combines the two features while attenuating, but not eliminating, their shared component. Specifically, for two features x and y with regression slope α (of y on x), an integrated variable is defined as

$$z = \frac{\alpha x + y}{1 + \alpha},$$

which is a convex combination weighted by the regression slope and reduces to the simple average when $\alpha = 1$.

Applying this construction to the present setting yields a two–stage integration. In the first stage, the most strongly correlated pair (b), (f) is integrated. The slope of the regression of (f) on (b) is $\alpha = 0.7276$ (Table 3), which is rounded to 0.7 for clarity. The intermediate variable $\tilde{b}$ is therefore defined as

$$\tilde{b} = \frac{0.7b + f}{0.7 + 1} = \frac{0.7b + f}{1.7},$$

where b and f are normalization values in features (b) and (f), respectively.

In the second stage, $\tilde{b}$ is taken as the consolidated representative of "recipe verbosity" (i.e., the joint information carried by character count and step count). Each of the remaining correlated features (c), (d), and (e) is then integrated with $\tilde{b}$ using the same construction. Because the regression slopes for (c), (d), and (e) in Table 3 are 1.9384, 0.9102, and 0.8263, respectively, these slopes are rounded to 1.9, 0.9, and 0.8 for the formulas. To preserve the convex–combination form of the construction, the slopes are placed on the side of $\tilde{b}$. The resulting intermediate variables are

$$\tilde{c} = \frac{1.9\tilde{b} + c}{1.9 + 1} = \frac{1.9\tilde{b} + c}{2.9},$$
$$\tilde{d} = \frac{0.9\tilde{b} + d}{0.9 + 1} = \frac{0.9\tilde{b} + d}{1.9},$$
$$\tilde{e} = \frac{0.8\tilde{b} + e}{0.8 + 1} = \frac{0.8\tilde{b} + e}{1.8},$$

where c, d and e are normalization values in features (c), (d) and (e), respectively.

Using these variables, four methods for quantifying recipe difficulty are proposed.

4.3.1 Method 1: All-Features Integrated Approach

$$Difficulty = \frac{a + \tilde{b} + \tilde{c} + \tilde{d} + \tilde{e}}{5}.$$

This method averages the cooking–action–based difficulty (a) and all intermediate variables. Because the impact of character count is included in each intermediate variable, recipes with longer character counts and more steps are expected to receive higher scores. However, there is a risk that the impact of character count may be overestimated.

4.3.2 Method 2: Average Model of $\tilde{c}$, $\tilde{d}$, and $\tilde{e}$

$$Difficulty = \frac{a + \frac{\tilde{c}+\tilde{d}+\tilde{e}}{3}}{2}.$$

This method averages the cooking–action-based difficulty (a) and the mean of the intermediate variables that reflect the number of cooking utensils, seasonings, and ingredients. Because it does not directly include $\tilde{b}$, the impact of the number of characters is smaller than in Method 1. Recipes with a large number of cooking utensils, seasonings, and ingredients are expected to receive higher scores.

4.3.3 Method 3: Average of All Intermediate Variables

$$Difficulty = \frac{a + \frac{\tilde{b}+\tilde{c}+\tilde{d}+\tilde{e}}{4}}{2}.$$

This method averages the cooking–action–based difficulty (a) and the mean of all intermediate variables. The results are expected to be similar to those of Method 2, but including $\tilde{b}$ slightly increases the influence of the number of steps.

4.3.4 Method 4: Simple Averaging Method

$$Difficulty = \frac{a + c + d + e}{4}.$$

This method averages the cooking–action–based difficulty (a) and the counts of cooking utensils, seasonings, and ingredients. It does not consider the number of characters or steps, making it the simplest aggregation method. Recipes with high cooking–action–based difficulty that require many utensils, seasonings, and ingredients will receive a high score. This is considered an intuitive and easy–to–understand evaluation method.

5 Experiment

In this section, the proposed method is verified through experiments.

5.1 Statistics of Difficulty Scores

Table 4 Statistics on Difficulty Scores for Each Method.

	Average	Median	Minimum	Maximum
Method 1	0.12	0.12	0.004	0.58
Method 2	0.20	0.21	0.003	0.57
Method 3	0.20	0.20	0.003	0.56
Method 4	0.19	0.19	0.008	0.61

The statistical measures for the difficulty scores computed using each method are shown in Table 4.

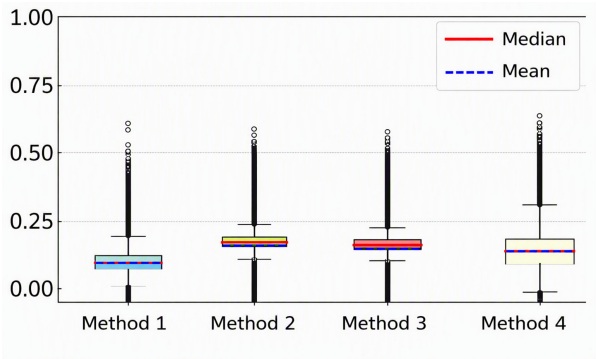


Fig. 1 Distribution of Difficulty Scores

Figure 1 shows the distributions of the difficulty scores obtained using the four methods as box–and–whisker plots. The vertical axis shows the difficulty score. The plots indicate that the scores span a wide range, confirming that the difficulty scores have been successfully subdivided. As shown in Table 4, the scores of Methods 2 to 4 are distributed in similar ranges, with medians of 0.21, 0.20, and 0.19, respectively, whereas the scores of Method 1 are concentrated in a lower range, with a median of 0.12. The maximum scores are 0.58, 0.57, 0.56, and 0.61 for Methods 1 to 4, respectively. Two factors explain the relative scarcity of recipes with scores above approximately 0.5. First, only 3.4% of the cooking–action level is rated at 4, so the action–based component (a) rarely takes its maximum value. Second, recipes posted on Cookpad are generally written for ordinary household cooking, so very difficult recipes are rare in the source data itself.

5.2 Evaluation Based on a Questionnaire Survey

This section describes the survey conducted to verify the validity of the four proposed difficulty computation methods.

5.2.1 Survey Method

The survey is conducted online, asking participants to rank nine recipes by difficulty.

The nine recipes evaluated in the survey are selected based on the following criteria: Five recipes are selected as those falling within the median range for each of the following metrics: number of characters, number of cooking utensils, number of seasonings, number of ingredients, and number of steps. The remaining four recipes are selected as those with the median difficulty score computed using the four proposed methods.

5.2.2 Survey Results

The ranking results obtained from 30 female respondents in their 20s are compiled, and the sum of ranks and the average rank for each recipe are computed. The results are shown in Table 5.

Table 5 Rank sums and average ranks for each recipe (survey results).

Rank	Recipe	Sum of ranks	Average rank
1	A	211.0	7.03
2	B	208.0	6.93
3	C	174.0	5.80
4	I	171.0	5.70
5	G	134.0	4.47
6	H	133.0	4.43
7	E	119.0	3.97
8	D	111.0	3.70
9	F	89.0	2.97

A higher average rank indicates that more respondents rated the recipe as "difficult". Recipe A is rated the most difficult, with an average rank of 7.03, while Recipe F is rated the easiest, with an average rank of 2.97.

To assess whether there are statistically significant differences in difficulty among the nine recipes based on the survey results, a Friedman test is conducted. The Friedman test evaluates the significance of rankings assigned by multiple individuals and is computed as follows.

$$\chi_r^2 = \frac{12(R_1^2 + R_2^2 + \cdots + R_k^2)}{nk(k+1)} - 3n(k+1).$$

Here, n represents the number of respondents ($n = 30$ in this study), k represents the number of comparison groups ($k = 9$ in this study), and R_j ($j = 1, 2, \cdots, k$) represents the sum of ranks assigned to the j-th recipe.

The results of the Friedman test are shown below.

- Friedman statistic: 66.00
- p–value: $p < 0.001$

Based on these results, the survey findings are considered statistically significant at the 0.1% level. Note that this test verifies that the nine recipes differ in perceived difficulty; it does not by itself establish the superiority of any particular computation method.

5.2.3 Similarity Evaluation for Rankings

Table 6 Comparison of Survey Results and Rankings for Each Method.

Rank	Survey results	Method 1	Method 2	Method 3	Method 4
1	A	A	B	B	A
2	B	B	A	A	B
3	C	I	I	I	E
4	I	E	E	E	H
5	G	C	C	C	C
6	H	F	H	G	F
7	E	H	G	F	I
8	D	G	F	H	G
9	F	D	D	D	D

The rankings based on the survey results and the difficulty scores computed using the four proposed methods are compared. Table 6 shows the rankings generated by each proposed method.

To quantitatively evaluate the similarity between the survey results and the rankings produced by each method, the following four metrics are used. For all of these metrics, a smaller value indicates that the two rankings are more similar.

The first is Spearman's footrule distance, defined as the sum of the absolute differences in rank for each feature.

$$F(\sigma, \tau) = \sum_{i=1}^{n} |\sigma(i) - \tau(i)| .$$

Here, $\sigma(i)$ and $\tau(i)$ are the ranks of recipe i in the two rankings, respectively. The minimum value is 0; when it is 0, the rankings are identical.

The second measure is the Kendall distance. This is the number of pairs for which the order is reversed between the two rankings. For all pairs (i, j), it counts the pairs where $i < j$ in one ranking but $i > j$ in the other.

The third is the Cayley distance. This is the minimum number of swaps required to make the two rankings match. The fourth is the Ulam distance. This is a distance based on the longest common subsequence.

Table 7 Rank–distance values between each method's ranking and the survey ranking. Smaller values indicate higher similarity.

	Spearman footrule	Kendall	Cayley	Ulam
Method 1	14	8	6	4
Method 2	12	7	5	5
Method 3	14	7	7	4
Method 4	18	10	6	3

The computation results for each metric are shown in Table 7. Method 2 shows the minimum value for three of the four metrics, indicating that it produces the ranking most similar to the survey results. Method 2 can be said to be the most balanced approach. In particular, the Footrule distance of 12 — the smallest value — means that the deviation in the rankings of each recipe is the smallest. Furthermore, since the Kendall distance is 7 and the Cayley distance is 5 — both small values — it can be confirmed that the overall structure of the ranking is very similar to the survey results. Note that these rank–distance values are descriptive comparisons: because only nine recipes are ranked, the differences among the four methods (e.g., a Footrule distance of 12 versus 14) are not tested for statistical significance.

5.3 Discussion

This section discusses the advantages of Method 2 based on the experimental results.

5.3.1 Comparison with the other methods

Because Method 1 directly includes $\tilde{b}$ (a composite variable combining character count and number of steps), there is a risk that the influence of character count is overestimated. In the correlation analysis (Table 2), the number of characters shows a very strong correlation ($r = 0.86$) with the number of steps; therefore, directly including $\tilde{b}$ carries the risk of double–counting information regarding the number of steps. By contrast, Method 2 is designed not to include $\tilde{b}$ directly but to incorporate it indirectly in the calculation of $\tilde{c}$, $\tilde{d}$, and $\tilde{e}$. This approach is believed to have appropriately mitigated the influence of the number of characters while maintaining a balance with other features. Method 2 uses the average values of three features: the number of cooking utensils ($\tilde{c}$), the number of seasonings ($\tilde{d}$), and the number of ingredients ($\tilde{e}$). These features directly reflect the complexity of the physical preparation involved in cooking.

- A large number of cooking utensils → Increased effort for preparation and cleanup
- A large number of seasonings → More complex seasoning
- A large number of ingredients → Increased effort for shopping and prep work

By averaging these features, it is believed that extremely skewed evaluations are avoided and a balanced assessment of difficulty is achieved.

The survey results, which rated Recipe A (12 types of seasonings, 14 ingredients) as the most difficult, support this interpretation. Although Recipe A does not have an unusually high number of steps (6), the complexity of preparation — due to the large number of ingredients and seasonings required — is presumed to have increased the perceived difficulty.

5.3.2 Reasons for the Superiority of Method 2

Method 1 incorporates all intermediate variables ($\tilde{b}$, $\tilde{c}$, $\tilde{d}$, $\tilde{e}$) together with (a). The Footrule distance is 14, the same as Method 3 and second only to Method 2 (12), but the Kendall distance is 8, which is somewhat large. The main issue is that the influence of character count becomes excessive. Since $\tilde{b}$ is included directly, recipes with a longer character count tend to receive excessively high scores. In fact, the Footrule distance is larger than that of Method 2, resulting in ranking discrepancies for some recipes.

Method 3 uses the mean values of all intermediate variables ($\tilde{b}$, $\tilde{c}$, $\tilde{d}$, $\tilde{e}$). The Kendall distance is 7, the same as in Method 2, but the Cayley distance is 7, the largest value. Method 3 is similar to Method 2, but the inclusion of $\tilde{b}$ slightly increases the influence of the number of steps. Since the Cayley distance is large, more swaps are required to match the rankings, suggesting that the structural similarity is somewhat lower. However, it demonstrates performance equivalent to Method 2 in terms of the Kendall distance and yields good results in terms of order reversal.

Method 4 is the simplest method, which involves normalizing the number of cooking utensils, seasonings, and ingredients and then simply averaging them together with (a). Since it does not take into account the number of characters or the number of steps at all, it is intuitive and easy to understand. However, its Footrule distance is 18 — the largest — and its Kendall distance is 10, which is also larger than that of the other methods. This method shows the greatest discrepancy from the survey results. In particular, the fact that it ranks Recipe I (11 steps) 7–th (on the easier side) is problematic. In the survey, Recipe I is rated 4–th (on the difficult side), suggesting that ignoring the number of steps leads to a significant error. The results of Method 4 suggest that the number of steps significantly affects difficulty, so it is not appropriate to ignore this feature. However, the Ulam distance is 3, the smallest value, indicating that the method demonstrates consistent performance from the perspective of the longest common subsequence.

5.3.3 Validity of Method 2

To verify the validity of Method 2, the survey results are compared with the rankings of the recipes for each feature. Table 8 below shows the rankings for each indicator: number of characters (b), number of cooking utensils (c), number of seasonings (d), number of ingredients (e), and number of steps (f). Entries shown in the same color within a single column indicate tied ranks. In the rankings generated by the proposed method, there are no ties. In contrast, the results for each category contain multiple ties.

Table 8 Rankings by each individual feature, compared with the survey ranking. Within each feature column, entries shown in the same color indicate tied ranks.

Rank	Survey	(b)	(c)	(d)	(e)	(f)
1	A	I	E	A	A	I
2	B	B	F	B	B	A
3	C	E	A	C	E	C
4	I	C	H	G	C	B
5	G	D	I	E	D	F
6	H	F	C	H	G	D
7	E	G	D	D	I	E
8	D	A	B	F	F	G
9	F	H	G	I	H	H

Table 9 Rank–distance values between each single–feature ranking and the survey ranking. Smaller values indicate higher similarity. Single–feature rankings contain ties, which are broken alphabetically by recipe label; the proposed method requires no tie–breaking.

Feature	Spearman footrule	Kendall	Cayley	Ulam
(b)	26	16	6	5
(c)	32	22	8	6
(d)	10	6	4	2
(e)	16	9	5	4
(f)	18	13	5	5

To compare similarity with the survey ranking, Spearman's footrule distance, the Kendall distance, the Cayley distance, and the Ulam distance are computed. Since the single–feature rankings contain ties, they are evaluated with ties broken alphabetically by recipe label. The results for each metric are shown in Table 9. Table 9 shows that the number of seasonings (d) and the number of ingredients (e) strongly reflect the perceived difficulty of the recipes. Method 2 predicts the perceived difficulty with high accuracy by using these two features as its core and complementarily integrating the number of cooking utensils, which is insufficient on its own.

The objective of this paper is to refine the difficulty classification. The cooking–action–based difficulty (a), based on References [15, 17], is unsuitable because many recipes are assigned a value of 1 or 2. As shown in Table 8, using each individual feature results in multiple recipes having tied rankings, making them unsuitable. The proposed method does not include tied rankings and thus provides a more refined classification. Therefore, the proposed method is appropriate.

6 Conclusion

This paper proposes a method for computing recipe difficulty from approximately 605,000 entries in the Cookpad dataset, taking into account six features: cooking

action level, number of characters, number of cooking utensils, number of seasonings, number of ingredients, and number of steps.

Four integration methods based on correlation and regression analysis are proposed, and their validity is confirmed through a survey of 30 participants. Among the four methods, Method 2 is found to be the most consistent with the survey results because it appropriately captures the physical complexity of cooking (the number of cooking utensils, seasonings, and ingredients) while avoiding overestimating the number of characters.

A limitation of this study is the scale and composition of the evaluation: the survey involves nine recipes and 30 respondents, all of whom are women in their 20s. The generalizability of the results to users of different ages, genders, and levels of cooking experience therefore remains to be verified. Expanding the survey to a larger and more diverse group of respondents, together with a statistical comparison of the integration methods, is left for future work.

The results of this study are expected to be applied to support systems that help users select recipes suited to their skill level.

Acknowledgments. This work was supported by Initiative for Realizing Diversity in the Research Environment. This work was supported by JST SPRING, Grant Number JPMJSP2115. The authors are grateful to the National Institute of Informatics for granting permission to use the Cookpad dataset.

Declarations

Funding

Initiative for Realizing Diversity in the Research Environment. JST SPRING, Grant Number JPMJSP2115.

Ethics approval

Not applicable.

Availability of data and materials

The data used in the paper can be obtained by submitting a request to Reference [1].

References

[1] National Institute of Informatics: Cookpad Data. Informatics Research Data Repository, https://www.nii.ac.jp/dsc/idr/cookpad/cookpad.html

[2] Harashima, J., Ariga, M., Murata, K., Ioki, M.: A large–scale recipe and meal data collection as infrastructure for food research. In: Proc. 10th International Conference on Language Resources and Evaluation (LREC 2016), pp. 2455–2459. ELRA, Portorož (2016)

[3] Wolfson, J.A., Bleich, S.N.: Is cooking at home associated with better diet quality or weight–loss intention? Public Health Nutr. **18**(8), 1397–1406 (2015). https://doi.org/10.1017/S1368980014001943

[4] Hartmann, C., Dohle, S., Siegrist, M.: Importance of cooking skills for balanced food choices. Appetite **65**, 125–131 (2013). https://doi.org/10.1016/j.appet.2013.01.016

[5] Vidgen, H.A., Gallegos, D.: Defining food literacy and its components. Appetite **76**, 50–59 (2014). https://doi.org/10.1016/j.appet.2014.01.010

[6] Lavelle, F., McGowan, L., Spence, M., Caraher, M., Raats, M.M., Hollywood, L., McDowell, D., McCloat, A., Mooney, E., Dean, M.: Barriers and facilitators to cooking from 'scratch' using basic or raw ingredients: a qualitative interview study. Appetite **107**, 383–391 (2016). https://doi.org/10.1016/j.appet.2016.08.115

[7] Min, W., Jiang, S., Liu, L., Rui, Y., Jain, R.: A survey on food computing. ACM Comput. Surv. **52**(5), 1–36 (2019). https://doi.org/10.1145/3329168

[8] Tran, T.N.T., Atas, M., Felfernig, A., Stettinger, M.: An overview of recommender systems in the healthy food domain. J. Intell. Inf. Syst. **50**(3), 501–526 (2018). https://doi.org/10.1007/s10844-017-0469-0

[9] Bondevik, J.N., Bennin, K.E., Babur, Ö., Ersch, C.: A systematic review on food recommender systems. Expert Syst. Appl. **238**, 122166 (2024). https://doi.org/10.1016/j.eswa.2023.122166

[10] Freyne, J., Berkovsky, S.: Intelligent food planning: personalized recipe recommendation. In: Proc. 15th International Conference on Intelligent User Interfaces (IUI 2010), pp. 321–324. ACM, New York (2010)

[11] Teng, C.-Y., Lin, Y.-R., Adamic, L.A.: Recipe recommendation using ingredient networks. In: Proc. 4th Annual ACM Web Science Conference (WebSci 2012), pp. 298–307. ACM, New York (2012)

[12] Trattner, C., Elsweiler, D.: Investigating the healthiness of internet-sourced recipes: implications for meal planning and recommender systems. In: Proc. 26th International Conference on World Wide Web (WWW 2017), pp. 489–498. IW3C2, Geneva (2017). https://doi.org/10.1145/3038912.3052573

[13] Mori, S., Maeta, H., Yamakata, Y., Sasada, T.: Flow graph corpus from recipe texts. In: Proc. 9th International Conference on Language Resources and Evaluation (LREC 2014), pp. 2370–2377. ELRA, Reykjavik (2014)

[14] Kiddon, C., Ponnuraj, G.T., Zettlemoyer, L., Choi, Y.: Mise en place: unsupervised interpretation of instructional recipes. In: Proc. 2015 Conference on

Empirical Methods in Natural Language Processing (EMNLP 2015), pp. 982–992. ACL, Lisbon (2015). https://doi.org/10.18653/v1/D15-1114

[15] Makino, N., Shioi, R., Kusunoki, K., Hatano, K.: Proposal of a difficulty factors method for recipe retrieval based on cooking actions. In: Proc. DEIM Forum 2017, C4-5 (2017) (in Japanese)

[16] Kusu, K., Makino, N., Shioi, T., Hatano, K.: Calculating cooking recipe's difficulty based on cooking activities. In: Proc. 9th Workshop on Multimedia for Cooking and Eating Activities (CEA 2017), pp. 19–24. ACM, New York (2017). https://doi.org/10.1145/3106668.3106673

[17] Iwamoto, J., Miyamori, H.: Proposal of a recipe retrieval system considering cooking difficulty. In: Proc. DEIM Forum 2012, E1-3 (2012) (in Japanese)

[18] Yajima, A., Kobayashi, I.: "Easy" cooking recipe recommendation considering user's conditions. In: Proc. DEIM Forum 2009, E5-6 (2009) (in Japanese)

[19] Müller, G., Bergmann, R.: Cooking made easy: on a novel approach to complexity-aware recipe generation. In: Proc. ICCBR 2017 Workshops (CAW, CBRDL, PO-CBR), Doctoral Consortium, and Competitions. CEUR Workshop Proceedings, vol. 2028, pp. 229–236. CEUR-WS.org (2017)

[20] Takahashi, R., Takahata, M., Kishi, N.: Estimating cooking difficulty level by clustering for recipe search. In: Proc. 20th Forum on Information Technology (FIT 2021), D-022, pp. 205–206 (2021) (in Japanese)

[21] Kagawa Education Institute of Nutrition: The Certificate of Home Cooking Skills, https://www.ryouken.jp/

[22] McCann, P. O.: Fugashi: A tool for tokenizing Japanese in Python. GitHub repository (2020), https://github.com/polm/fugashi

[23] Kudo, T.: MeCab: Yet Another Part-of-Speech and Morphological Analyzer (2005), https://taku910.github.io/mecab/