Recommendation System for Complementary Breastfeeding using Ontology Modelling and Naïve Bayes

S W Sihwi¹, A N Fadhilah¹, M P Puspasari¹, and Winarno¹

¹ Department of Informatics, Faculty of Mathematics and Natural Science, Universitas Sebelas Maret, 57126, Indonesia

Abstract. Complementary breastfeeding is an additional food given to the baby started from six to 24 months. The giving of complementary foods is given gradually according to the age of the children and adapted to the condition of the children, such as allergies suffered or malnutrition suffered. This research aims to develop an ontology based decision support system that will help mothers in giving breastfeeding to their babies with keep regarding to their food preferences. This research successfully develops a content-based recommendation system by performing the Naïve Bayes Method and modified the existing ontology modelling and also develop a mobile application with the Android platform so that it can be more accessible to many people. Thirty-five users evaluated the system, and the result of the usability testing shows that the user satisfaction level using SUS (System Scale Usability) method is 79.57, which is in Grade A-. This grade indicates that the system can be well understood by the user and can help mothers in choosing breastfeeding recipes or menus.

1. Introduction

When the baby enters the age of 6 months, consuming breast milk alone will be not enough for the baby's nutritional needs and energy. As a consequence, complementary food is needed by the babies in that ages [1]. The complementary breastfeeding food (MPASI) is given to infants during the weaning period, which is from the age of 6 months to 2 years [1], without leaving at all the breast milk as nutrition and protective factors for the disease until the child reaches the age of two years.

Several studies focusing on giving MPASI recipes have been carried out, as in the research conducted by [2] which integrates ontology modelling with the MPASI menu domain [3] and children's nutritional needs domain [4]. In the study conducted by [2], it provides some choices of MPASI menu that are appropriate to the condition of the child. It also gives the baby’s needed nutrition based on the knowledge from the ontology.

There are some advantages are able to gain in [2] by using ontology. One of them is it will not only store the data but it will also describe the concepts in the domain and explains the relationships that are in it [5]. Besides that, ontology also makes researchers easier to share knowledge because it uses common vocabulary in domain explanations. Ontology itself can be defined as "an explicit specification of a conceptualization" [6]. Technically, the representation of ontology is in the form of objects, properties of objects, and relations between each object [6]. Ontology is represented by using the language of the OWL (Ontology of Web Languages). In the beginning, the OWL is designed to represent information about the categories of an object and how the object relates. OWL can also provide information about the object itself [6].

However, from the study of [2], there are some weaknesses. First, it is unable to learn about the user behaviours and preferences. It is because the resulting menu choices coming from the ontology filtering, and it is also because no history of recipes that are chosen by the user, so the resulting menu
choices are less varied. Second, it was build in the desktop apps platform that can not be easier for users to access it compared to if it was build on a mobile apps platform. With the growth and popularity of smartphones that are increasing year by year around the world with smartphone user statistics of 1 million and market growth of approximately 42%, so the use of the native app in mobile is universal [7]. According to [7], the native app is an application installed on a smartphone. In addition, the number of smartphone users in Indonesia as of January 2018 reached 177.9 million and the number of internet users reached 132.7 million from the population of Indonesia which reached 265.4 million [8], so that this study will be made in the form of mobile applications with the Android platform to make it easier use by the society.

To overcome with the shortcomings of [2], this research will develop a content-based recommendation system using MPASI ontology modelling that has been carried out by [2] in the form of mobile applications with the Android platform to be more accessible to many people with the help of internet connections. A content-based recommendation system will recommend items that are similar to what the user previously liked. The value of similarity between items is calculated based on the features available on each content. As an example in the movie domain, if a user gives a positive rating on a comedy genre film, the system can recommend items from that genre [9].

There are many algorithms that can be used in content-based recommendation system. One of the most effective and efficient classification algorithms is Naïve Bayes Classifier [10]. Even though it is a simple algorithm, it has a high level of accuracy and is relatively fast when faced with massive data [11]. It assumes that the effect of an attribute in a class is independent of the value of another attribute, where this assumption is called independence condition class which makes computing in it simpler so that it can be called naïve [12]. However, even when the independence assumption is violated, there is still apparent accuracy of it [10]. Hence, this research uses Naïve Bayes Classifier.

There are several studies related to ontology modelling, recommendation systems, and Naive Bayes methods that have been carried out, as in the research conducted by [13] who made English-language news classifications using the ontology to handle the word synonyms. Besides that, there is also a research that created an ontology modelling in the health domain to look for the relationship of symptoms and diseases was utilized by performing the probability of each relation in the ontology [14]. Another study was conducted by [15] who made a social networking-based recommendation system to find medical-related sites in Tunisia using ontology with a domain of user interest and medical-related places in Tunisia.

2. Naïve Bayes Classifier

Naïve Bayes Classifier is a simple probabilistic classification method rooted in the Bayes Theorem and assumes all attributes are independent or not interdependent with other attributes [12]. Naïve Bayes Classifier assumes that the effect of an attribute in a class is independent of the value of another attribute, where this assumption is called independence condition class which makes computing in it simpler so that it can be called naïve [12]. The classification process requires some clues to determine what class is suitable for the sample analysed. Therefore, the Bayes theorem is adjusted as follows [16]:

\[
P(C|F_1,\ldots,F_n) = \frac{P(C)P(F_1,\ldots,F_n|C)}{P(F_1,\ldots,F_n)}
\]

(1)

Where variable \(C\) represents the class, while variable \(F_1, \ldots, F_n\) represents the characteristics of the clues needed to carry out the classification. That formula explains the chance of entry of a sample with specific attributes in the class \(C\) (posterior) is the chance of emergence of the class \(C\) (before the entry of the sample, often called prior), multiplied by the probability of the appearance of sample characteristics in the class \(C\) (likelihood), divided by opportunities emergence of global sample characteristics (evidence). The evidence value always fixed for each class in one sample. The value of the Posterior will later be compared with the Posterior values of the other classes to determine the class of a sample will be classified [16].
3. Research Methodology

3.1. Data collection
In this work, the ontology data from [2] is used without changing the structure of the ontology. However, ontology [2] had an insufficient number of instances (menus and ingredients) and also because it did not considered ingredients causing allergies. As consequences, in this stage, its data was increased by collecting other data taken from several sources, such as internet sources [17][18], and book [19]. In adding the data, the local wisdoms in Indonesia are considered. The recommended MPASI menu does not contain ingredients such as pork, dog meat, blood, or other ingredients that are not commonly consumed by most Indonesians.

3.2. System analysis and design.
At this stage, the first step is analyzing the existing system to get option MPASI menu in [2] in order to look for strengths and weaknesses from that system. Based on the results of the analysis on the existing system, this study will develop a system that has been made before by taking the advantages and make some adjustments to overcome the weaknesses of the system. This study used the ontology modelling conducted by [2]. Besides that, this research used Naïve Bayes method and content-based recommendation with feedback likes or dislikes for the chosen recipes. This study also added the data in the form of ingredients that have risk of causing allergies.

3.3. System development
This system was build using PHP ((Hypertext Preprocessor) programming language. Tere are some tools, framework and libraries used to develop this system. They were Protégé 5.1 to build the ontology model, Ionic v1.0 framework to build it in the Android platform, ARC2 to be able to perform ontology queries in PHP (Hypertext Preprocessor), and MySQL to store the user information.

3.4. System evaluation
System evaluation is done by evaluating usability using the SUS method. The system is tested and evaluated by the users that are women having babies aged 6-24 months and also familiar with MPASI. Besides that, they also have to familiar with gadgets.

The method used for usability evaluation is System Usability Scale (SUS) method. John Brooke first introduced the SUS method in 1986, and in its use, ten statements are distinguished in odd numbers in the form of positive statements, while even numbers are negative statements on a scale of 1-5 [20]. The list of statements for evaluating user satisfaction with the SUS method is shown in Table 1. The evaluation criteria use a scale of 1 to 5, which has 1 as the lowest value and 5 as the highest value.

| No | Code | Statements |
|----|------|------------|
| 1  | P1   | I think that I would like to use this application frequently. |
| 2  | P2   | I found the application unnecessarily complex. |
| 3  | P3   | I thought the application was easy to use. |
| 4  | P4   | I think that I would need the support of a technical person to be able to use this application. |
| 5  | P5   | I found the various functions in this application were well integrated. |
| 6  | P6   | I thought there was too much inconsistency in this application. |
| 7  | P7   | I would imagine that most people would learn to use this application very quickly. |
| 8  | P8   | I found the application very cumbersome to use. |
| 9  | P9   | I felt very confident using the application. |
SUS Score can be obtained using the following formula:

\[
SUS\ Score = \sum ((odd\ _\ number - 1) + (5 - even\ _\ number)) \times 2.5
\]

(2)

SUS Score has a range of values from 0-100. To find out the quality grade of the application, this SUS score should be mapped to Table 2, the Sauro-Lewis Curved Grading Scale.

| SUS Score Range | Grade | Percentile Range |
|-----------------|-------|------------------|
| 84.1 - 100      | A +   | 96 - 100         |
| 80.8 - 84.0     | A     | 90 - 95          |
| 78.9 - 80.7     | A -   | 85 - 89          |
| 77.2 - 78.8     | B +   | 80 - 84          |
| 74.1 - 77.1     | B     | 70 - 79          |
| 72.6 - 74.0     | B -   | 65 - 69          |
| 71.1 - 72.5     | C +   | 60 - 64          |
| 65.0 - 71.0     | C     | 41 - 59          |
| 62.7 - 64.9     | C -   | 35 - 40          |
| 51.7 - 62.6     | D     | 15 - 34          |
| 0.0 - 51.6      | F     | 0 - 14           |

4. Result and Discussion

4.1. Data collection

The data collected was 70 recipes and 22 ingredients that added to existing ontology modelling. So, the total is 141 recipes and 121 ingredients.

4.2. System analysis and design

4.2.1. Adding data property. In this work, we modified the existing ontologies [2], namely by adding data property. The addition of data property `mengandungResiko` aims to increase knowledge that in a recipe there is the ingredient that is susceptible to allergens. To use data properties properly, need annotation property as a property link with related individuals. Therefore, there are additional annotation properties, namely `data_mengandungResiko`.

Allergies arise due to changes in the reaction of an ingredient in the surrounding environment. Allergy is a reaction given by the body to a component that is harmless because the body's defenses are not reliable and cannot stand [22]. About 2% of adults, 10% of infants under one year of age, and 4-8% of children up to 5 years of age have food allergies [17]. Everyone can have allergies to certain foods. There are eight food ingredients that can cause allergy [17] and [19], namely: wheat, tree nuts, shellfish, chicken eggs, cow's milk, fish, soybeans, and sesame seeds. So that in the built system also applies filters to food ingredients contained in a recipe. So that if it turns out, there are ingredients that included in the category of allergens, the user will be given a notification.

4.2.2. Adding instance. At first, the number of recipes was only 71, and the number of ingredients was 99 in the ontology. However, after adding the number of recipes to 141 and the number of ingredients to 121.
4.2.3. System architecture. The system architecture as shown in Figure 1 has three layers, which are Data layer, PHP REST API layer and Android App layer. The Data Layer is covered all data and knowledge that is used in the application including in MySQL and also in ontology. The data will be used as input and knowledge of the application. PHP REST API layer, which is a web service, that is used to exchange of data using HTTP as a protocol for data communication. The REST Server will provide resources in the form of JSON. This resource will appear in the REST Client which is the android application. So if the user take action in the application, the Android layer will send API Request, where this API Request is input in the PHP REST API MPASI layer. Next, the PHP REST API MPASI layer will process by sending requests to the data layer. The data layer will give the response in the form of data needed and the PHP REST API layer will process the data and will generate resources with the JSON format. This resource will be translated by the Android layer so that the user can see the results of the actions taken. This process will continue every time the user takes action in the Android application.

![Figure 1. System architecture](image)

4.2.4. System activity diagram. Figure 2 and Figure 4 show all the processes in the new system as a new user (Figure 2) and also as an old user (Figure 4).

![Figure 2. System activity diagram for a new user](image)

If the user is a new user, then the user must fill in the child's profile first. Next, if the user add ingredients have been tried before, then the system will display recipes recommendation with ingredients have been tried. If the user did not add ingredients have been tried, the system will display recommendation recipes with a single ingredient. Also, the user can choose recipes based on the name of the ingredients. The both of choices using filters such as allergy ingredients, malnutrition, age, containing new ingredients <= 1.
If the user is an old user, then the system will check whether she has been more than three days not choose the recipe recommendations and give feedback to the system. If it is more than three days, then the user must update the child's profile and must give feedback on the recipe (if there are recipes that have not been given feedback). If it is less than three days, then the user must give feedback on the recipe (if there are recipes that have not been given feedback). Furthermore, to display the menu recommendations, the user can choose the recommendation menu based on the chosen ingredient or not. The recommendation menu is generated using filters such as allergy ingredients, malnutrition, age, containing new ingredients $\leq 1$. However, if the user selects a recommendation menu based on selected ingredients, then one more filter is added, namely the chosen ingredient and Naïve Bayes is also implemented. Whereas if the recommendation menu is without the selected ingredient, the user can choose recipe recommendations using the four-days rule or Naïve Bayes. The four-days wait rule is a rule that introduces one type of food for four days to see how the baby reacts to a type of food. This rule needs to be applied because the reaction to several new types of food will arise within 24 hours, while for digestive problems it takes longer [23].

4.3. System development

4.3.1. Naïve Bayes. This algorithm is used to get the babies’s preferences. To make the system development of the algorithm clearer, an example was made. For example, a user who has 13 months old baby without having malnutrition and has given feedback on eight chosen recipes (without using four-days rule) as can be seen in table 3. In the next recommendation, she will get recommendations containing a list of recipes as many as 44 recipes, for display of recommendations, can be seen in Figure 6.

Table 3. Example of chosen recipes history

| Recipe     | Ingredient | Feedback |
|------------|------------|----------|
| Puree Persik | Peach      | like     |
Based on the results of the recommendations above (figure 5), the stewed apple recipe ranks first because the highest probability value belongs to it. Here is the calculation for the stewed apple recipe.

$$P(liked) = \frac{\text{the number of to be liked recipes}}{\text{the number of all recipes}} = \frac{7}{141} = 0.0496453 \quad (3)$$

$$P(apple) = \frac{\text{the number of apples in the to be liked recipe} + 1}{\text{the number of ingredients in the to be liked recipe} + \text{the number of all ingredients}} = \frac{2+1}{9+121} = 0.0230769 \quad (4)$$

$$P(\text{Stewed Apple}) = P(liked) \times P(apple) = 0.0496453 \times 0.0230769 = 0.0011456 \quad (5)$$

4.3.2. Normalization. This normalization is used to normalize the value of feedback given on a new ingredient entered by the user after more than three days not giving feedback. It can be seen for more details in Figure 7. Normalization of values is needed so that the value of giving feedback to a new ingredient that is entered by the user becomes balanced when compared to the amount of feedback value of each ingredient has been tried by the user. In addition, because it is possible for a new ingredient to be mixed with other ingredients, so to make it fair and balanced, value normalization is needed. Giving feedback that is entered by the user will be normalized by taking the minimum value and the maximum value of the amount of feedback value (the amount of feedback each ingredient that has been tried by the user). The following are the formulas used to normalize.
Where, \( x' \) is the normalization value, \( x \) is the feedback value that entered by the user, \( \text{min value} \) is the minimum value of the amount of feedback value, and \( \text{max value} \) is the maximum value of the amount of feedback value. The amount of feedback value is the amount of feedback each ingredient that has been tried by the user. For example, a user has more than three days not opened the application. When opening the app, the user will fill out a form such as a profile page (Figure 6) to update the profile and update the ingredient that contains the name of the new ingredient and feedback on giving the new ingredient (Figure 7).

In the ingredient feedback section, the user is asked to provide feedback on a new ingredient that has been tried. The user is asked to choose one of several feedback options, as in Figure 7. Each feedback option has its own value, namely: disliked = 1, rarely preferred = 2, sometimes preferred = 3, preferred = 4, and always preferred = 5. Scale score 1-5 is chosen because it simplifies the choice of user feedback on a new ingredient that has been tried. If the scale value is too much, it will make too many feedback choices, making it difficult for the user to choose feedback. However, there are conditions that the value will enter directly without being normalized first, namely when the minimum value is = 0, the maximum value = 0, and the maximum value = 1. It is done to avoid the 0 value and if a user has a minimum value = 1 and the maximum value = 1 in the amount of feedback value for each ingredient that has been tried. Because if the minimum value and maximum value = 1, then after normalization, any feedback options that are chosen by the user, it will produce a value such as the value of the chosen feedback option.

\[
x' = \frac{x \times \text{min value}}{\text{max value}}
\]

(6)

**Table 4.** The example of normalization value
| Ingredient | Feedback ingredient | The amount of feedback each ingredient that has been tried by the user | Normalization Result |
|------------|---------------------|-------------------------------------------------|---------------------|
| apricot    | sometimes preferred | peach = 2, apple = 4, banana = 2, oat = 1, tempeh = 1, strawberry = 1, avocado = 2, peas = 1 | $x' = \frac{x \times \text{min value}}{\text{max value}}$ |

Normalization Result

| Ingredient | Normalization Result |
|------------|---------------------|
| apricot    | So, the value for apricot is 1 |

4.4. System testing

System evaluation is done by evaluating the system usability. The system is tested and evaluated by 35 users, namely mothers who are familiar with gadgets and have babies aged 6-24 months or mothers who are familiar with MPASI. Usability evaluation using the SUS method uses ten statements (Table 1) with a scale of 1 to 5. By obtaining the SUS Score which is resulted from equation 3, application grade are easily translated from Table 2. The results of usability testing can be obtained by using the following formula:

$$
Usability = \frac{\sum \text{SUS Score}}{\text{number of responden}} = \frac{2785}{35} = 79.57
$$

(7)

An application is considered to be good enough if its SUS Score more than equal to 68 [19], while based on the results of usability testing, in this application the average SUS Score is 79.57. Based on Table 2, SUS Score 79.57 falls in the value range of 78.9 - 80.7, which is in Grade A-.

5. Conclusion

In this work, we successfully develop a new system which utilized the advantages and the weaknesses of the previous research which built the MPASI Decision Support System with ontology modelling. Some adjustments have been performed by adding instances and data properties to the existing ontology. We also develop a mobile application for the proposed system with the Android platform that uses ontology modelling and the Naïve Bayes method. By performing the proposed method, it is possible for the MPASI menu to deliver the recommendation base on the child's profile. Finally, the evaluation which uses SUS method shows that the usability of the proposed system is 79.57, which is in Grade A-.

6. References

[1] Deavara, Yoga. (2016). Memberikan Makan pada Bayi: Kapan, Apa, dan Bagaimana?. [online] http://www.idai.or.id/artikel/klinik/pengasuhan-anak/memberi-makan-pada-bayi-kapan-apa-dan-bagaimana [Accessed 3 Jan. 2017].

[2] Paramitha, Sihwi, S.W., Anggrainingsih (2018). Decision Support System of Complementary Breastfeeding Based on Ontology Modelling. International Seminar and Research of Information Technology and Intelligent System (ISRITI). Yogyakarta, Indonesia.

[3] Sihwi, S.W, Athiyah, Doewes,A. (2016). Ontology Model for Breastfeeding Complementary Recipes. The 3rd International Conference on Information Technology, Computer and Electrical Engineering (ICITACEE 2016). Semarang, Indonesia.

[4] Sari, D. I., Sihwi, S., & Anggrainingsih, R. (2014). Ontology Model in Pediatric Nutrition Domain. International Conference on Information Technology Systems and Innovation (ICITSI). Bandung.

[5] Sachs, Eliza. (2006). Getting started with Protégé-frames. BMJ Clinical Research Ed, 307(6919), 1619-1620.
[6] Afuan, L and Azhari. (2016). Penerapan Semantik Web pada Ontologi Learning Resource Repositori. Seminar Nasional Teknologi Informasi dan Komunikasi 2016 (SENITKA 2016) ISSN: 2089-9815.

[7] Jobe, W. (2013). Native Apps vs. Mobile Web Apps. International Journal of Interactive Mobile Technologies (iJIM), 7(4), 27-32.

[8] Sidiq, F. (2018). Pengguna Perangkat Mobile di Indonesia Semakin Tinggi, Ini Datanya!. [online] http://industri.bisnis.com/read/20180201/101/733037/pengguna-perangkat-mobile-di-indonesia-semakin-tinggi-ini-datanya [Accessed 8 Aug. 2018].

[9] Ricci, F., Rokach, L., Shapira, B., & Kantor, P. B. (2011). Recommender Systems Handbook. New York: Springer Science & Business Media.

[10] Al-Aidaroos, K.M, Bakar, A.A, Othman, Z. (2010). Naïve bayes variants in classification learning. 2010 International Conference on Information Retrieval Knowledge Management (CAMP). Selangor, Malaysia.Han, Jiawei. (2006). Data Mining: Concepts and Techniques Second Edition. San Francisco: Morgan Kaufmann Publisher. Elsevier Inc.

[11] Han, Jiawei. (2006). Data Mining: Concepts and Techniques Second Edition. San Francisco: Morgan Kaufmann Publisher. Elsevier Inc.

[12] Patil, T. R., & Sherekar, S. S. (2013). Performance Analysis of Naive Bayes and J48 Classification Algorithm for Data Classification. International Journal of Computer Science and Applications, 6(2), 256-261.

[13] Muflikhah, L. and Murdianto, A. (2016). Classification of News Document in English Based on Ontology. British Journal of Applied Science & Technology, 17(4), pp.1-9.

[14] Sandun, I., Sumathipala, S., & Ganegoda, G. (2017). Self-Evolving Disease Ontology for Medical Domain Based on Web. International Journal of Fuzzy Logic and Intelligent Systems, 17(4), pp.307-314.

[15] Frihka, M., Mhiri, M., and Gargouri, F. (2015). A Semantic Social Recommender System Using Ontologies Based Approach For Tunisian Tourism. ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal, 4(1), p.90.

[16] Bustami, B. (2014). Penerapan Algoritma Naïve Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. Jurnal Informatika, 8(1).

[17] Santacruz, Sally. (2016). Top 8 Most Common Food Allergies. [online] https://www.foodsafety.com.au/resources/articles/top-8-most-common-food-allergies[Accessed 16 Jan. 2018].

[18] CDC (Center for Disease Control and Prevention). (2018). Food Allergies in Schools. [online] https://www.cdc.gov/healthychildren/foodallergies/index.htm [Accessed 16 Jan. 2018].

[19] Pistiner, M., LeBovidge, J., & Food Allergy Canada. (2015). Living Confidently With Food Allergy 2nd Edition Handbook. Food Allergy Canada.

[20] Sauro, Jeff. (2011). Measuring Usability with The System Usability Scale (SUS). [online] https://measuringu.com/sus/ [Accessed 28 May. 2018]

[21] Lewis, J. (2018). Measuring Perceived Usability: The CSUQ, SUS, and UMUX. International Journal of Human–Computer Interaction, pp.1-9.

[22] Candra, Y., Setiarini, A., & Rengganis, I. (2011). Gambaran Sensitivitas Terhadap Alergen Makanan. Makara Journal of Health Research, 44-50.

[23] Momtastic. The 4 Four Day Wait Rule - Using the 4 Day Wait Rule, to Test for Baby Food Allergies in Introducing Solid Foods. [online] http://wholesomebabyfood.momtastic.com/4daywaitrule.htm [Accessed 28 May. 2017].