Dietary Recommendation Based on Recipe Ontology

Yu-Hsien Ting, Qiangfu Zhao
Computer and Information Systems
The University of Aizu
Fukushima, Japan
{m5172101,qf-zhao}@u-aizu.ac.jp

Rung-Ching Chen
Department of Information Management
Chaoyang University of Technology
Taichung, Taiwan
crching@cyut.edu.tw

***Abstract*—It is known that unhealthy diet, irregular life, work pressure and other factors can result in a number of diseases. Diabetes mellitus, peptic ulcer, and gastro-enteritis are just a few examples. To reduce these diseases, diet management is becoming an indispensable part in our daily lives. The main purpose of this study is to build a diet management system that can provide the user's correct nutritional information based on recipe ontology. In this paper, we construct a recipe ontology that defines various food nutrients needed for healing some common diseases, and an inference engine that can, given the user's health conditions as well as his/her preferences, recommend a proper recipe based on the recipe ontology.**

Keywords—recipe; ontology; dietary recommendation; disease

I. Introduction

With the advance of medical science and technology; increase in personal income; and improvement in living standards, our life span has been increased significantly. However, unhealthy diet, irregular life style, work pressure and other factors, can result in a number of diseases. Diabetes mellitus, Cardiovascular, peptic ulcer and gastro-enteritis are just a few examples. In fact, people are now spending more for medical care than ever before. According to the Ministry of Health of Japan [14], the prevalence of these common diseases is increasing gradually, and the average ages of people suffering these diseases are decreasing. In order to increase the quality of living and to reduce the expenses for medical care, it is important to understand the dietary type and suitable intake, and to avoid uneven or inappropriate diet.

The main purpose of health care is to achieve a balanced and healthy diet in our daily lives, and to reduce the chances to visit hospitals. In recent years, diet management is becoming an important topic, and is now an indispensable part of our daily lives. The Ministry of Education, Culture, Sports, Science and Technology of Japan is building a Food Composition Database website [5] that provides nutrients of food query. Using this website, users can know the nutrients intake, and may make a balanced diet each day. However, the website does not maintain long-term personal diet records, and thus cannot provide appropriate diet information for each user.

In this study, in order to provide better nutritional information for each user, we propose a recipe ontology (RO) which is built based on SPARQL Protocol and RDF Query Language (SPARQL); and a dietary recommendation system (DRS) based on the above RO and a JENA Semantic Web Framework (JENA). Our design goal is to make this DRS able to analyze whether a user has a balanced diet based on the user's own situation, and recommend suitable diet recipes for the user.

For users suffering from some diseases but in an early stage, he/she can use the DRS system to find out the relationship between diet and diseases, and the DRS system can recommend suitable dietary plan based on the user's personal needs and preference.

II. Related work

In this section, we first provide a brief survey on dietary recommendation, and then introduce some fundamental knowledge related to this research.

A. Related work

Recipe recommendation has been an important topic for many years. Generally speaking, recipe recommendation is based on the user's history record [12][13][16]. Geleijnse et al. presented in [6] a prototype of a personalized recipe advice system, which can suggest recipes based on the user's history, which includes past selection and nutrient intake. From the history record, it is possible to know whether a user's diet or nutrients intake is balanced or not.

How to achieve health through diet is a relatively new research topic. In [2], Chiu-Ming Hu built a recipe ontology based on common food recipes and a food composition database. Using this ontology, it is possible to establish nutritional and food information related for recipe, provide nutrient information, and calculate nutrients intake and dietary record. This approach only constructs the recipe and nutrients ontology, and does not suggest to the users personalized information, include influence of nutrients relationship between recipes.

In [1], Lo proposed a method for finding the most suitable types of dietary health therapy based on the user's own dietary needs or expectations of suffering a certain disease. This method allows users to reach a balanced and healthy diet goal, and thus can be used to improve the user's own dietary health care therapy. However, this method does not record the user's diet status, and does not manage long-term record of each user.

In [3], Su designed a Diet-Aid web service based on health screening data of Health Level Seven International (HL7), according to the user's health status, given appropriate dietary recommendations. The paper proposed a personalized recipe based on availability of personal nutritional needs, and a long-term record. However, this approach has not presented how to prevent the diseases of nutrients relationship between recipes.

B. Ontology

In computer science and information science, an ontology formally represents knowledge as a hierarchy of concepts within a domain, using a shared vocabulary to denote the types, properties and interrelationships of those concepts [15]. An ontology is a description (like a formal specification of a program) of the concepts and relationships that can formally exist for an agent or a community of agents. This definition is consistent with the usage of ontology as set of concept definitions, but more general [21].

Ontologies are often equated with taxonomic hierarchies of classes, class definitions, and the subsumption relation, but ontologies need not be limited to these forms. Ontologies are also not limited to conservative definitions, that is, definitions in the traditional logic sense that only introduce terminology and do not add any knowledge about the world. To specify a conceptualization, one needs to state axioms that do constrain the possible interpretations for the defined terms [15].

Common components of ontologies include classes, attributes, relations, individuals, rules and axioms. A class can be defined as a concept. Attributes are properties of objects in a class. Some attributes can be used to describe relationships between individuals. Relations connect different classes or individuals to each another. Instances are the basic objects, and through properties connect to classes. Rules use if-then sentences to describe the logical reasoning. Axioms are specific logical assertions.

```
PREFIX abc: <http://example.com/exampleOntology#>
SELECT ?capital ?country
WHERE {
  ?x abc:cityname ?capital ;
     abc:isCapitalOf ?y .
  ?y abc:countryname ?country ;
     abc:isInContinent abc:Africa .
}
```

Fig. 1. An example of SPARQL program [20]

C. SPARQL

SPARQL is a recursive acronym for SPARQL Protocol and RDF Query Language. SPARQL essentially consists of a standard query language for retrieving and manipulating data stored in Resource Description Framework format. It was made a standard by the RDF Data Access Working Group (DAWG) of the World Wide Web Consortium [10]. SPARQL queries hide the details of data management, which reduces costs and increases robustness of data integration on the Web. “Trying to use the Semantic Web without SPARQL is like trying to use a relational database without SQL,” explained Tim Berners-Lee, W3C Director [7]. Therefore, SPARQL has become an official W3C Recommendation.

SPARQL uses query SELECT to extract RDF/OWL repository. It can also use the CONSTRUCT query to construct new triples from existing ones, so it is different from the SQL. An SPARQL query example that models the question "What are the capitals of all countries in Africa?" is shown in Fig. 1.

Variables are indicated by a "?" or "$" prefix. The data set are bindings for ?capital and the ?country will be returned. In the query, the prefix "abc" stands for “http://example.com/exampleOntology#” [20].

D. Jena

Jena is an open source Semantic Web framework for Java. It provides an API to extract data from and write to RDF graphs. The graphs are represented as an abstract "model". A model can be sourced with data from files, databases, URLs or a combination of these. Jena provides support for OWL (Web Ontology Language). The framework has various internal reasoners and the Pellet reasoner (an open source Java OWL-DL reasoner) can be set up to work in Jena [9].

The Jena API fully implements the RDF specification. The Jena Architecture is shown in Fig. 2.

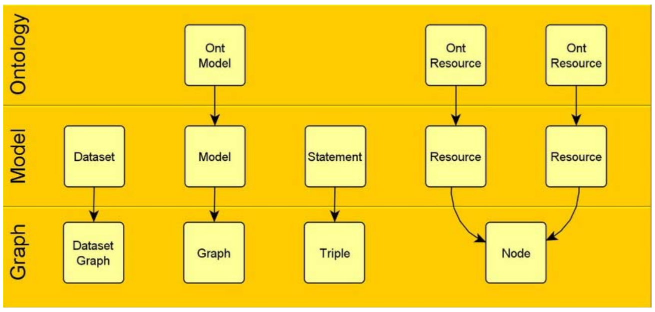

Fig. 2. Jena architecture [4]

From the above figure we can see that the Graph layer is the base layer which serves as an SPI (Service Provider Interface). If we want to use Jena in a product, we must extend graph interfaces. The Model layer creates a triple, and it is an API (Application Provider Interface). Most developers leverage Jena support in their business applications with the model layer. The final layer is the ontology layer, which is an inference API. The ontology layer provides the inference capabilities. That is, the ability to work with triples that are implied, in addition to the pre-defined triples.

III. Dietary Recommendation System

This study proposes a dietary recommendation system (DRS) that can analyze whether a user has a balanced diet based on the user's own situation, and recommend suitable diet recipes for the user. This DRS can be used by different users and each user can obtain his/her personal recommendations. The system architecture diagram is shown in Fig. 3.

The recommendation process of the DRS system is divided into five steps as follows:

1) Recipes are extracted from two recipe sources website [18] [19]. These websites provide the specified recipe.

2) A Health and Nutrition database is built based on data extracted from the Food Composition Database (FCD) [5] and source of Japan Preventive Association of Life-style related Disease (JPALD) [8]. This database provides various nutrients, an adult intake nutrient needed in a day and calculate the amount of nutrient to eat in one meal.

3) The ontology is constructed from Recipe, FCD and JPALD, and this ontology defines the relationship between recipe to nutrients and diseases. The relationship not only provides recipe queries, but also indicates a balanced diet of nutrients intake, and the way for prevention of life style-related disease [11].

4) The recipe recommendation interface uses PHP, HTML and MYSQL to construct a website. The users, after sign-in and providing some limited personal information related to his/her preference and health condition, can obtain recommendations. A long-term diet record is used to manage the health status of each user.

5) User's query is converted using ontology to SPARQL queries, and this is used to query the recipe ontology. The DRS then makes a recommendation through JENA inference.

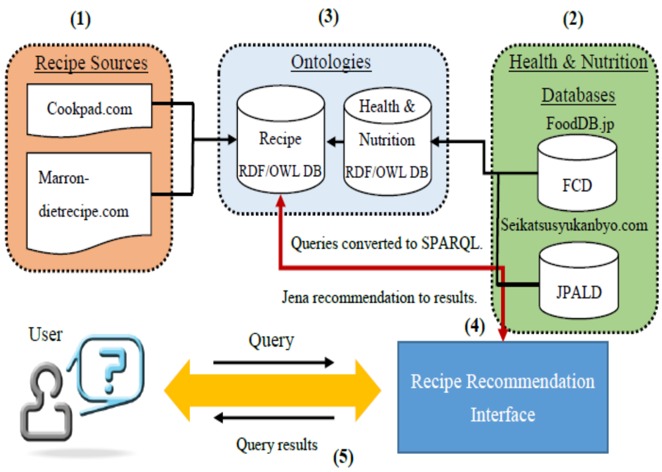

Fig. 3. Dietary recommendation system architecture

A. *Knowledge Base in Ontology used Protégé*

Protégé was developed by Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine. Protégé is a free, open-source platform that provides a growing user community with a suite of tools to construct domain models and knowledge-based applications with ontologies [17].

The knowledge base is built by using the Protégé software editor, in which the ontology tree was extracted using RDF/OWL (Fig. 4). In our system, the ontology root directory is divided into three categories, namely recipe, food and diseases. The recipe subclass has 14 categories. The attributes of recipe include the recipe name, cooking methods, meal times, food, calories, and so on. The food subclass can be classified in 18 categories based on Food Composition Database website built by the Ministry of Education, Culture, Sports, Science and Technology of Japan. The attributes of the food subclass include weight and nutrition, and the attribute values for real instances.

The disease subclass in the current prototype system has only two diseases to recommend. One is diabetes mellitus, and the other is hypertension. In the future we will add more diseases. Hypertension is one of the 10 leading causes of death in Japan. Diabetes mellitus is one of the top 10 leading causes of death for women in Japan.

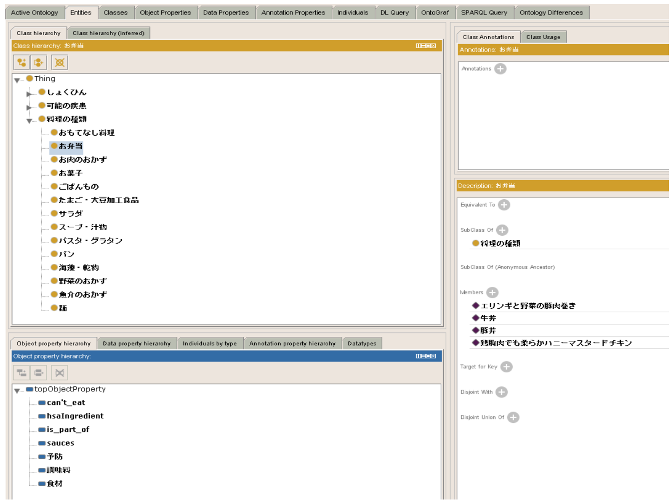

Fig. 4. Protégé software editor interface

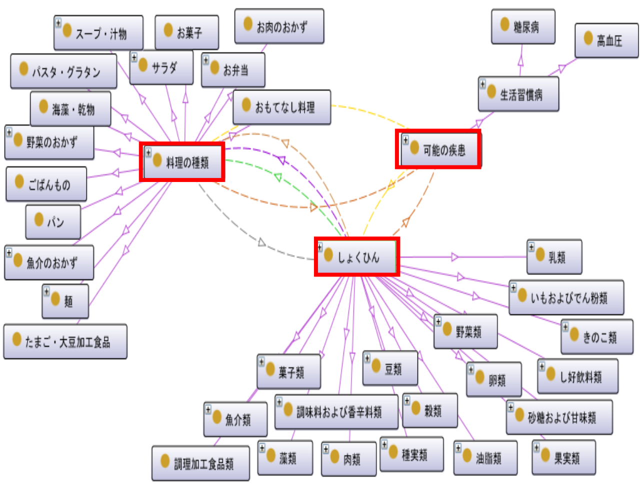

Fig. 5. Schematic diagram of ontology relationships

The DRS knowledge base has three subclasses, namely recipe, food and diseases as Fig. 5. The food subclass is related with recipe by “is part of”, and reversely recipe is related with food by “has ingredients”. Recipe and food are related to the disease subclass via “prevent”, and reversely, disease is related to recipe and food via “suitable”.

B. Knowledge Base retrieval and recommendation

The knowledge base retrieves recipes that matches the user preferences and the given user data, and recommends suitable diet. In this study, we use SPARQL and JENA to provide personalized recommendations. The first thing to do is to load the knowledge base, and the user's personal data from the user profile and the dietary record. Then the system uses a series of SPARQL queries to generate a personalized diet. Using the results of SPARQL, the system finds out and recommends the most suitable recipe for the user based on JENA rules.

The Jena rules are defined based on the user's disease, preference and dietary record. These rules can be used to find the recipes that fit the user's current condition. The system selects the user data from the database, gets recipes from the ontology, and determines the user's disease using the SPARQL queries, and then uses Jena if-then rules to determine a diet for recommendation.

The prefix "recipe" stands for "http://www.semanticweb.org/jocelyn/ontologies/2014/1/Recipe#." Basically, SPARQL query produces all recipes for a given recipe type. If the user has some disease, SPARQL query will add the condition, produces recipes suitable for this disease, and filter out other recipes. An example of SPARQL query is shown in Fig. 6.

```
if(dis.equals("no")) {
    queryString =
            "PREFIX recipe:<http://www.semanticweb.org/jocelyn/ontologies/2014/1/Recipe#>"+
            "PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>"+
            "PREFIX owl: <http://www.w3.org/2002/07/owl#>"+
            "SELECT * WHERE {" +
            "?料理の種類 recipe:料理の名稱 ?x ."+
            "}";
} else if(dis.equals("diabetes mellitus,hypertension")) {
    queryString =
            "PREFIX recipe:<http://www.semanticweb.org/jocelyn/ontologies/2014/1/Recipe#>"+
            "SELECT * WHERE {" +
            "?料理の種類 recipe:群尿? ?y ."+
            "?料理の種類 recipe:料理の名稱 ?x ."+
            "FILTER(?y = \"糖尿病.高血圧\")"+
            "}";
} else if(dis.equals("diabetes mellitus")) {
    queryString =
            "PREFIX recipe:<http://www.semanticweb.org/jocelyn/ontologies/2014/1/Recipe#>"+
            "SELECT * WHERE {" +
            "?料理の種類 recipe:群尿? ?y ."+
            "?料理の種類 recipe:料理の名稱 ?x ."+
            "FILTER(?y = \"糖尿病\")"+
            "}";
} else  {
    queryString =
            "PREFIX recipe:<http://www.semanticweb.org/jocelyn/ontologies/2014/1/Recipe#>"+
            "SELECT * WHERE {" +
            "?料理の種類 recipe:群尿? ?y ."+
            "?料理の種類 recipe:料理の名稱 ?x ."+
            "FILTER(?y = \"高血圧\")"+
            "}";
}
```

Fig. 6. SPARQL query of determine the disease

The Jena rules defined in this study are if-then rules for selecting recipes. Below are just some examples:

- If [disease is null and preference is salty and record is null] Then recommend 5 recipes at random from salty recipes.
- If [disease is null and preference is sweet and record is null] Then recommend 5 recipes at random from sweet recipes.
- If [disease is null and preference is spicy and record is null] Then recommend 5 recipes at random from spicy recipes.
- If [disease is null and preference is acid and record is null] Then recommend 5 recipes at random from acid recipes.
- If [disease is diabetes mellitus and preference is salty and record is null] Then recommend 5 recipes that fit diabetes mellitus from salty recipes.
- If [disease is diabetes mellitus and preference is salty and record is not null] Then recommend 5 recipes that fit diabetes mellitus and user's record from salty recipes.

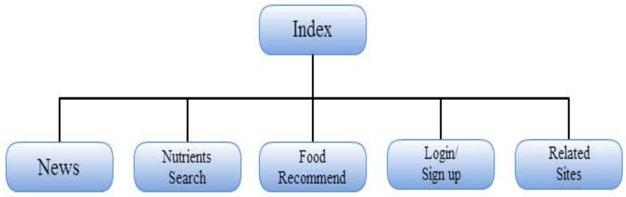

Fig. 7. DRS basic site structure

IV. Implementation

The DRS is built on Microsoft Windows Professional 7. The system uses MySQL database to record user history. The system server is Apache Server. PHP, HTML, and JavaScript are used to construct the system website. Protégé is used to develop the ontology, and the ontology is used along with the Eclipse-based JENA.

The DRS interface includes news, nutrients search, food recommendation, login/sign up and related sites (Fig. 7). In the News page, a "recipe of day" is posted for everyone. Nutrients search page provides food nutrients from which users can know how many nutrients they should obtain each day. The food recommendation page includes three sections, namely general, prevention diseases, and dieting. The user can choose one to fit his/her current situation. Through signup and checking some items, the users can get better dietary advices and recommendations to fit their preferences. The basic site structure of the DRS is shown in Fig. 7.

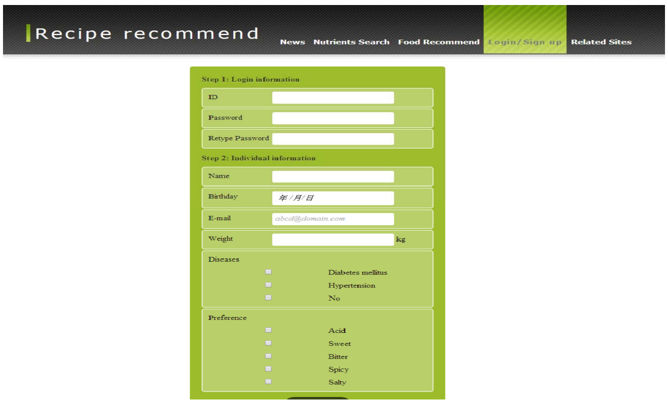

Fig. 8. DRS user registration interface

The user registration page is shown in Fig. 8. The user needs to fill in the some limited personal information, e.g.

name, birthday, email, preference and disease. After login the user can obtain recipe recommendation, information related to nutrient and graphs for visualizing the information. Generally speaking, managing a long-term health condition is useful to prevent diseases, to achieve a balanced diet, and to reduce chances of visiting hospitals (Fig. 11). In the personal record page, there is a record of the user's past diet, and the user can download a PDF file at any time.

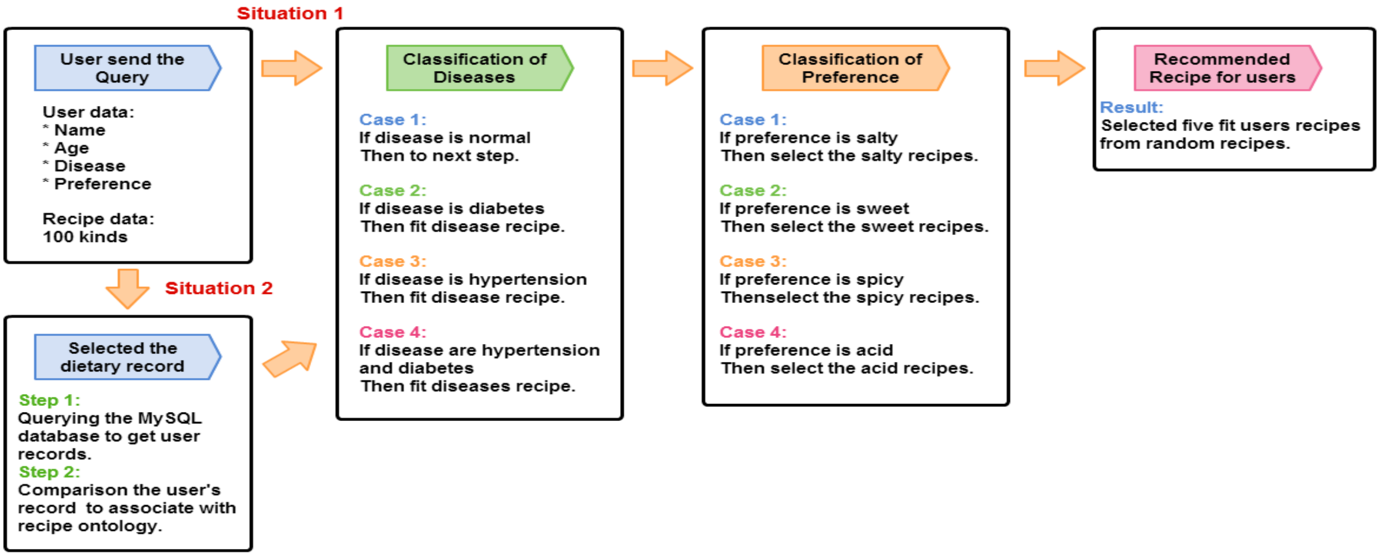

Fig. 9. DRS recommend flow chart

In this study, the user health status is divided into four cases, namely normal, diabetes mellitus, hypertension, and both of diseases. The four cases are as follows:

- If the health status is normal, the user just wants to obtain recipe recommendation and keep a balanced diet.
- If the health status is diabetes, the user needs some recipe to heal the diabetes and achieve a balanced diet.
- If the health status is hypertension, the user needs some recipe to heal hypertension and achieve a balanced diet.
- If the health status is diabetes and hypertension, the user needs some recipe to heal diabetes and hypertension, then achieve a balanced diet.

The process of Jena recommendation can be divided into four phases as shown in Fig. 9. In the first phase, the user sends a query to DRS along with the user data, which include name, age, preference, and disease; and the system obtains a set of recipes from the database through search. In the second phase, DRS filters out recipes that are not good for the current health status. In the third phase, the system filters out recipes that do not match the user's preference. In the last phase, the system recommends 5 recipes from the set of recipes obtained in the previous phases.

userid	name	birthday	Weight	e-mail	preference	disease
test	Bill	1983-06-01	50	bill@yahoo.com	Salty	Diabetes mellitus

Fig. 10. A example of User data from database.

For example, suppose that a user named Bill login to the system. His personal information is shown in Fig. 10. Bill sends the query to DRS. He likes to eat salty food, and he may suffer from diabetes mellitus. Therefore the DRS recommends recipes that are salty and fit the disease. The recommended result is shown in Fig. 11.

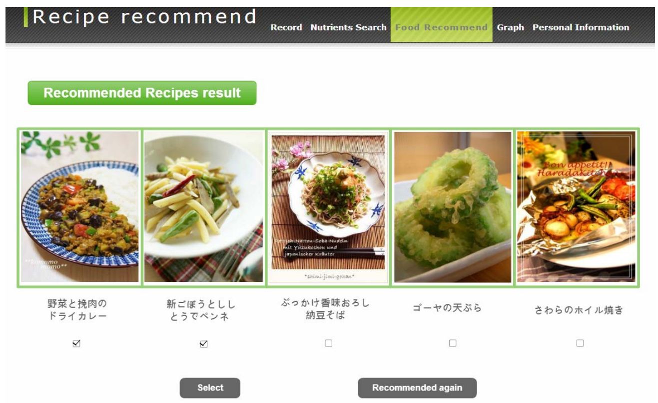

Fig. 11. Recipe recommended result.

V. Conclusion

In this paper, we have studied design of an ontology-based dietary recommendation system (DRS) that can analyze whether a user has a balanced diet based on the user's own situation, the user disease, preference and the dietary record; and can recommend suitable dietary recipes for the user. Users can query individual dietary records from the history, and also can download the dietary record when necessary.

The current prototype system consists of a recipe ontology (RO) which includes SPARQL Protocol and RDF Query Language (SPARQL); and an inference engine based on JENA Semantic Web Framework (JENA). The main purpose of this study is to consider the personal health information and give users personal dietary recommendations, to improve the user's physical condition, and management a long-term personal dietary record.

In order to make the proposed more practically useful, it is needed to collect a lot of recipes from different databases; to improve the website contents and extend the website functionality; to recommend with high accuracy; to increase SPARQL query ability; and to provide suitable JENA recommendation rules.

In addition, we will try to extend the ontology architecture, using user dietary record to construct the personal recipe ontologies. A personal ontology can be used locally in a user terminal, and can be useful even if the network is not available. This is important not only in cases of artificial/natural disasters, but also for protecting user's privacy. We would like to solve the problems step by step in the future.

REFERENCES

[1] Chi-Chun Lo, Chi-Hua Chen, Chih-Chien Lu, and Ding-Yuan Cheng, "The Design and Implementation of a Curative Food Recommendation Mechanism Using Chinese Food Therapy as a Case Study," Journal of e-Business, vol. 14, no. 3, pp. 513-548, 2012.

[2] Chiu-Ming Hu, "A nutrition analysis system based on recipe ontology," University of Taipei Medical, 2006.

[3] Chuan-Jun Su, Chia-Wen Chih and Yin-An Chen, "Ontology-based Personalized Diet Plan Web Service Using HL7 Health Screening Data," Proceedings of the Asia Pacific Industrial Engineering & Management Systems Conference, pp. 1164-1176, 2012.

[4] Craig Trim, "A Semantic Web Framework," *http://trimc-nlp.blogspot.jp/2013/06/introduction-to-jena.html*, June 6, 2013.

[5] "Food Composition Database website," *http://fooddb.mext.go.jp*

[6] Gijs Geleijnse, Peggy Nachtigall, Pim van Kaam, and Luciënne Wijgergangs, "A personalized recipe advice system to promote healthful choices," IUI '11 Proceedings of the 16th international conference on Intelligent user interfaces, pp. 437-438, 2011.

[7] Ivan Herman, "W3C sematic web," *http://www.w3.org/blog/SW/2008/01/15/sparql_is_a_recommendation*, January 15, 2008.

[8] "Japan Preventive Association of Life-style related Disease," *http://seikatsusyukanbyo.com*

[9] "Jena (framework)," *http://en.wikipedia.org/wiki/Jena_(framework)*

[10] Jim Rapoza, "SPARQL Will Make the Web Shine. eWeek," *http://www.eweek.com/c/a/Application-Development/SPARQL-Will-Make-the-Web-Shine/*, May 2, 2006.

[11] "lifestyle related disease," *http://ja.wikipedia.org/wiki/%E7%94%9F%E6%B4%BB%E7%BF%92%E6%85%A3%E7%97%85*

[12] Martin Svensson, Kristina Höök, and Rickard Cöster, "Designing and evaluating kalas: A social navigation system for food recipes," ACM Transactions on Computer-Human Interaction (TOCHI), vol. 12, no. 3, pp. 374-400, 2005.

[13] Mayumi Ueda, Mari Takahata, and Shinsuke Nakajima, "User's food preference extraction for personalized cooking recipe recommendation," Proc. of the Second Workshop on Semantic Personalized Information Management: Retrieval and Recommendation, vol. 781, no. 6, pp. 98-105, 2011.

[14] "Ministry of Health of Japan," *http://www.mext.go.jp/*

[15] "Ontology (information science)," *http://en.wikipedia.org/wiki/Ontology_(information_science)*

[16] Peter Forbes and Mu Zhu, "Content-boosted matrix factorization for recommender systems: Experiments with recipe recommendation," Proceedings of the fifth ACM conference on Recommender systems, pp. 261-264, 2011.

[17] "Protégé official website," *http://protege.stanford.edu*

[18] "COOKPAD," *http://cookpad.com*

[19] "料理レシピ," *http://marron-dietrecipe.com/category/category.html*

[20] "SPARQL," *http://en.wikipedia.org/wiki/SPARQL*

[21] Tom Gruber, "What is an Ontology," *http://www-ksl.stanford.edu/kst/what-is-an-ontology.html*,1992.