A Food Recommendation System Based on Semantic Annotations and Reference Prescriptions

Devis Bianchini(✉), Valeria De Antonellis, and Michele Melchiori

Department of Information Engineering, University of Brescia, Via Branze, 38, 25123 Brescia, Italy
{devis.bianchini,valeria.deantonellis,michele.melchiori}@unibs.it

Abstract. Food recommendation, as well as searching for health-related information, presents particular characteristics unlike conventional recommender systems, since it often has educational purposes, to improve behavioural habits of users. In this paper, we present a menu generation system that uses a recipe dataset and annotations to recommend menus according to user's preferences. Moreover, reference prescription schemes are defined to guide our system for suggesting suitable choices. Firstly, relevant recipes are selected by content-based retrieval, based on comparisons among features used to annotate both users' profiles and recipes. Then, menus are generated using the selected recipes and are ranked taking into account also prescription schemes.

1 Introduction

Recommendation systems find information of interests, properly customized according to the users' own preferences [1]. This is valid also for specific application domains, such as health and nutrition, where any choice made upon automatically provided recommendations might have an impact on users' health and wellness. Existing approaches for recommending food and health-related information consider four main aspects, namely personal and cultural preferences, health and religion constraints [2] to implement a food recommendation approach. Nevertheless, to the best of our knowledge educational purposes have not been taken into account yet. Personalized Health Information System (PHIRS) [3] is a recommendation system for health information that matches the user's profile against the retrieved health information, without considering culture and religion in the profile. CarePlan [4] is a semantic representation framework for healthcare plans that mixes the patients' health conditions, personal preferences, the medical knowledge and clinical pathways, but ignores other aspects, such as personalization coming from educational health information, user's culture and religion, that impact on the food choice. Same limitations affect the system described in [5]. Other systems do not address personalization at all, such as the HealthFinland project [6], a smart semantic portal that helps the users to find relevant health information using simple keywords instead of

M.A. Jeusfeld and K. Karlapalem (Eds.): ER 2015 Workshops, LNCS 9382, pp. 134–143, 2015.
DOI: 10.1007/978-3-319-25747-1_14

medical vocabularies. This variety of approaches demonstrates that users' profiling, in particular for sectors and domains such as the food and health recommendation, is mainly addressed in an ad-hoc manner, without aiming at providing some educational effect on the users.

In this paper, we present a menu generation system that uses a recipe dataset and annotations to recommend menus. Given the specific characteristics of food recommendation systems, reference prescription schemes are used to guide our system for suggesting suitable choices. Firstly, relevant recipes are selected by content-based retrieval, based on comparisons among features used to annotate both users' profiles and recipes. Then, candidate menus are generated, using the selected recipes, and are ranked also taking into account reference prescription schemes. The system is being developed and tested within a food recommendation regional project funded in Lombardy region, Italy[1]. Therefore, the contribution of this paper mainly relies on the innovative recommendation method, that is *education-oriented*, aiming at satisfying both user's preferences and reference prescriptions.

The paper is organized as follows: Sect. 2 provides detailed definitions about the recommendation model; in Sect. 3 we describe the menu generation procedure; Sect. 4 discusses preliminary experimental results; finally, in Sect. 5 we sketch conclusions and future work.

2 Recommendation Model

Let's consider, for example, Jasmine, who is looking for recipe suggestions to have lunch during her working hours. Jasmine is registered within a food recommender system, where she has an associated profile. Jasmine declared a preference for having meat during lunches. She suffers from long-term diseases, such as diabetes and high-blood-pressure, therefore white meat is more advisable. She belongs to the Islamic religion, so recommendations about any food that contains alcohol or pork are not acceptable, since this food is prohibited to Muslims. Other characteristics emerge if she is looking for recipes to cook at home, for which ingredients and cooking procedures assume a specific relevance. Each factor may be represented through a feature, that in turn might assume different values. Features can be used both to characterize the recommended items (e.g., recipes, dishes) and to represent users' profiles. Moreover, feature matching can be based either on the feature values contained in a request for suggestions by the user (short-term feature matching), or on the frequency of feature values contained within the history of past choices made by the user (long-term feature matching, aiming at considering long-term user's preferences). Existing food recommendation web sites[2] and approaches do not consider some important aspects that could be exploited for recommendation purposes. Firstly, recipes can be combined into different menus, but not all aggregations are suitable. Specific combinations of recipes might be due to particular menu configurations (e.g.,

[1] The Smart BREAK project, http://www.smartbreakproject.it.
[2] See for example http://www.food.com/, http://allrecipes.com.

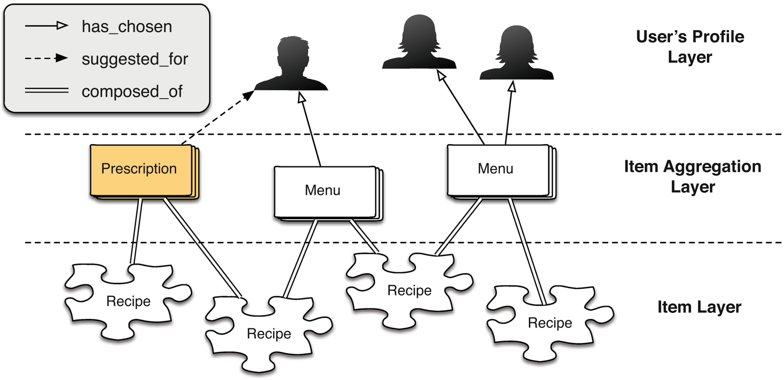

Fig. 1. The multi-layered framework adopted for food recommendation.

appetizer, first course, second course, dessert). Secondly, new challenges raise within application domains (food recommendation, healthcare, wellness), that present the particular features highlighted above. For instance, some Jasmine's preferences (e.g., having meat during lunches, all the days throughout the week) may contrast with best habits, according to up-to-date medical prescriptions. Therefore, recommendations of recipes also might be suggested in accordance with recent medical prescriptions, that recommend variety and balancing of different ingredients and nutrients, which recipes are composed of. This means that reference prescriptions might be used as first class citizens in recommending recipes to users who present particular profiles. Nevertheless, prescriptions cannot be totally imposed to users, disregarding their own preferences. Lifestyle improvements should gradually guide users towards better choices.

To this aim, the recommendation model we propose in this paper is based on the multi-layered framework shown in Fig. 1 and detailed in the following. Each layer is focused on specific elements, namely items to be recommended (i.e., recipes), item aggregations (menus, prescriptions) and users, further described with proper features and relationships with elements of the other layers.

The Item Layer. In this layer, recipes represent the most fine-grained items to be recommended. According to our general model, a recipe is described as $r_i = \langle ID_i, n_i, \mathcal{C}_i, \mathcal{T}_i \rangle$, where: ID_i ($\forall i = 1, \ldots N$) is the unique identifier of the recipe (we denote with $\mathcal{R}$ the overall set of N recipes available within the dataset); n_i is the name of the recipe; $\mathcal{C}_i$ and $\mathcal{T}_i$ are sets of features used to characterize the recipe. We distinguish among: (i) categories ($\mathcal{C}_i$), that classify the recipe and are taken from top-down domain-specific ontologies; (ii) semantic tags, that is, bottom-up keywords assigned to the recipes by users to annotate them and semantically disambiguated using a general-purpose lexical system or thesaurus, to face polisemy (that is, the same tag refers to different concepts) and synonymy problems (i.e., the same concept is pointed out using different tags), that traditional tagging may present. For semantic definition of categories, we extended the `food.owl` ontology[3]. For example, in our approach each recipe is classified through categories directly related to food, such as the `RecipeType` (e.g., appetizer, first course, second course, fruits, dessert), the `CookingStyle` (e.g., Asian cuisine), the ingredients used in the recipe (e.g., chicken, beef, rice), and other categories such as the `Religion` (e.g., Islamic) for which the recipe is meant, or the `Pathology` (e.g., diabetes, high-blood pressure), for which the recipe is advised. In Fig. 2 eight different recipes are depicted, with categories extracted from the ontology partially shown on the left. In our approach, semantic tagging is supported using the semantic disambiguation system extensively described in [7], where a semantic tag $t \in \mathcal{T}_i$ is a triplet, composed of: (i) the tag itself extracted from WordNet; (ii) the set of all the terms in the synset; (iii) the human readable definition associated with the synset.

The Item Aggregation Layer. In this layer, recipes are aggregated to be proposed in a combined way. In the context of our food recommendation approach, we distinguish two kinds of aggregations: (a) available *menus*, that is, combinations of recipes chosen in the past by the users of the system (these menus are used to extract the preferences of the users, exploiting them during the recommendation phase, see next section for details); (b) *prescriptions*, that is, proper combinations of recipes that are advisable for specific kinds of users. Formally, we define an aggregation (either a menu or a prescription) $a_j \in \mathcal{A}$ as $a_j = \langle n_{a_j}, \mathcal{R}[a_j], \tau_{a_j} \rangle$, where: $\mathcal{A}$ denotes the overall set of aggregations; n_{a_j} is the name of the aggregation; $\mathcal{R}[a_j] \subseteq \mathcal{R}$ is the set of recipes aggregated in a_j; τ_{a_j} is the template of the aggregation, expressed in terms of specific categories. In our approach, given an aggregation a_j, τ_{a_j} is identified considering the `RecipeType` category. Examples of templates may be [appetizer, first course, second course, dessert] or [first course, fruit]. Templates will play an important role for the formulation of the request for suggestions and to speed up the generation of the recommendation output (see Sect. 3). The way prescriptions are associated with users depends on the features used to describe users' profiles. In our food recommendation approach, Food Frequency Questionnaires (FFQ) are issued to collect users' habits and BMI (Body Mass Index), in order to automatically classify users within specific phenotypes [8], for which prescriptions have been inserted within the system. This task is supervised by medical doctors, who participate to the Smart BREAK project (see Sect. 4). The point here is that prescriptions are given and will be used, as shown in the next section.

The User's Profile Layer. In this layer, users are profiled according to their preferences and past menu choices, that are collected to represent the history of recipe and menu selections made by the user. Formally, we define the profile $p(u)$ of a user $u \in \mathcal{U}$ as $p(u) = \langle ID_u, \mathcal{C}[u], \mathcal{T}[u], \mathcal{M}[u], \mathcal{P}[u] \rangle$, where: $\mathcal{U}$ denotes the overall set of users; ID_u is used to identify the user u; $\mathcal{C}[u]$ and $\mathcal{T}[u]$ are the sets of features (namely, categories and tags) used to denote the preferences of u; $\mathcal{M}[u]$ is the set of menus chosen by the user in the past, that in turn may

[3] http://krono.act.uji.es/Links/ontologies/food.owl/view.

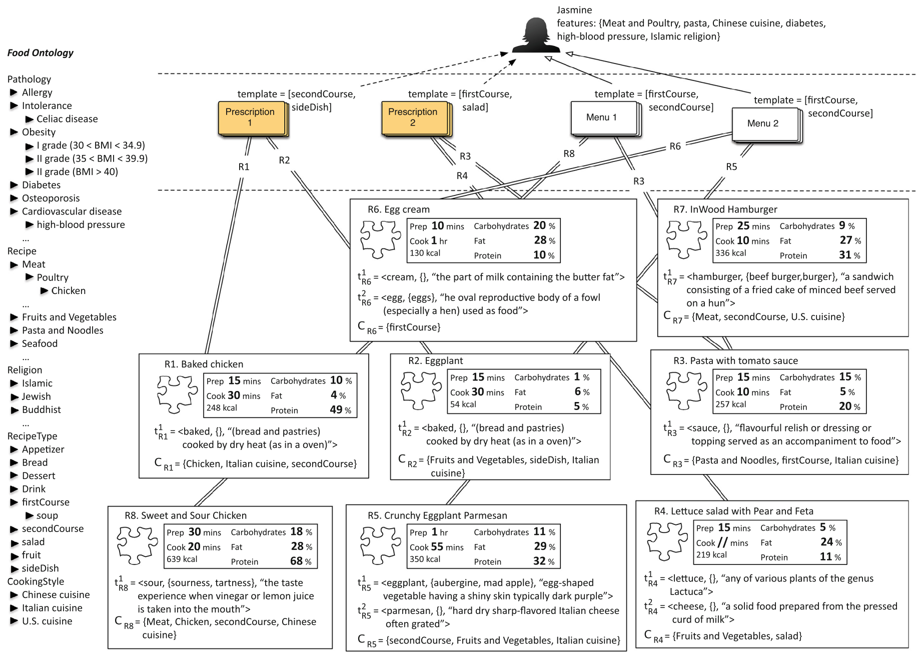

Fig. 2. Items to recommend (recipes) and aggregations (menus and prescriptions) of the running example.

represent the preferences of the user u about recipes to be recommended; $\mathcal{P}[u]$ is the set of prescriptions assigned to the user in the system. To characterize user's profiles, we rely on the classification features (i.e., categories and tags), whose values represent long-term preferences of the user, that might be collected and updated using traditional techniques from the literature [1].

3 Menu Recommendation System

When Jasmine is looking for menu suggestions, she generates a request $r_r(u)$ formulated as $r_r(u) = \langle \mathcal{C}_r, \mathcal{T}_r, \tau_r \rangle$, where: $\mathcal{C}_r$ is a set of categories that represent immediate, short-term preferences of Jasmine; similarly, $\mathcal{T}_r$ is a set of (semantic) tags, specified by issuing the request; τ_r is the menu template Jasmine is searching for. The recommender system takes into account the profile of the user u (Jasmine), that is, $p(u)$ whom the request comes from. To this aim, the request $r_r(u)$ is expanded with the categories and semantic tags that are present within the Jasmine's profile $p(u)$. We denote with $\widehat{r}_r(u)$ the expanded version of the request, where $\widehat{r}_r(u) = \langle \widehat{\mathcal{C}}_r, \widehat{\mathcal{T}}_r, \tau_r \rangle$. The set $\widehat{\mathcal{C}}_r$ contains both the categories specified in $\mathcal{C}_r$ and the categories within $p(u)$. The set $\mathcal{C}_r$ might also be empty, thus denoting that the system should exclusively rely on the preferences contained within $p(u)$. Each category $c_r \in \widehat{\mathcal{C}}_r$ is weighted by means of a coefficient $\omega_r \in [0,1]$ such that: (a) $\omega_r = 1$ if $c_r \in \mathcal{C}_r$, (b) $\omega_r = freq(c_r) \in [0,1]$ otherwise. The value of ω_r means that a category explicitly specified in the request will be considered the most for identifying candidate recipes. The term $freq(c_r)$ computes the frequency of category c_r among all the categories that annotate the recipes contained in the profile $p(u)$. Less frequent categories will be considered as less important for identifying candidate recipes. If a category c_r is present both in $\mathcal{C}_r$ and in the profile, then $\omega_r = 1$. The same applies for (semantic) tags. If u is a new user, without a history of past choices, then $\widehat{r}_r(u) = r_r(u)$ (no expansion). In this case, prescriptions are used to differentiate the user's choices, based on the user's phenotypes, as explained in the following. Frequencies are computed on a menu basis, since recipes are recommended only within menus. For instance, let's consider the recipes and Jasmine's profile shown in Fig. 2, and the following request, issued to search for menus and recipes containing `baked poultry`, according to [`firstCourse`, `secondCourse`] template: $r_r(u) = \langle\{\texttt{poultry}\}, \{\texttt{baked}\}, [\texttt{firstCourse}, \texttt{secondCourse}]\rangle$. The following expanded version of the request is generated (frequency values are specified among parenthesis):

$$\widehat{\mathcal{C}}_r = \{\texttt{poultry}(1.0), \texttt{meat}(0.5), \texttt{chicken}(0.5), \texttt{secondCourse}(1.0), \texttt{Chinesecuisine}(0.5), \texttt{PastaandNoodles}(0.5), \texttt{firstCourse}(0.5), \texttt{Italiancuisine}(1.0), \texttt{FruitsandVegetables}(0.5)\}$$

$$\widehat{\mathcal{T}}_r = \{\texttt{baked}(1.0), \texttt{sour}(0.5), \texttt{cream}(0.5), \texttt{egg}(0.5), \texttt{eggplant}(0.5), \texttt{parmesan}(0.5)\}$$

Feature-Based Recipe Filtering. The input of this step is the set $\mathcal{R}$ of all the available recipes and the request $\widehat{r}_r(u)$. First, τ_r element specified in the request is considered. Those recipes such that their `RecipeType` is not included within τ_r will not pass the feature-based filtering step. In the example above, only the `R1`, `R3`, `R5`, `R6`, `R7` and `R8` recipes will be further considered, that is, only recipes that are either first courses or second courses. Not all features can be exploited in the same way to filter out not relevant recipes. For instance, let's consider some constraints imposed by the Islamic religion or by some allergies. Recipes that do not respect these constraints must be excluded before any other kind of comparison. These constraints, to keep our model as more general as possible, are defined within the domain ontology and are expressed in terms of other features. For example, the Islamic religion within the Jasmine's profile excludes all recipes that are annotated with `pork` or `alcohol` as contained ingredients. Modeling of such constraints must be accurate; this explains why we inserted them within the domain ontology, that is developed in a controlled way. After τ_r and ontological constraints have been used to pre-select recipes, the filtering based on remaining features is applied, according to the following similarity metrics.

Category-Based Relevance. The relevance of a recipe $r_i = \langle ID_i, n_i, \mathcal{C}_i, \mathcal{T}_i \rangle$ with respect to the request $\widehat{r}_r(u) = \langle \widehat{\mathcal{C}}_r, \widehat{\mathcal{T}}_r, \tau_r \rangle$ taking into account categories in $\mathcal{C}_i$ and $\widehat{\mathcal{C}}_r$, denoted with $Sim_{cat}(\widehat{r}_r, r_i) \in [0,1]$, is computed as:

$$Sim_{cat}(\widehat{r}_r, r_i) = \frac{2 \cdot \sum_{c_r, c_i} \omega_r \cdot CatSim(c_r, c_i)}{|\mathcal{C}_i|} \in [0,1] \tag{1}$$

where c_r ranges over the set $\widehat{\mathcal{C}}_r$, c_i ranges over the set $\mathcal{C}_i$, $|\mathcal{C}_i|$ denotes the number of categories in the set $\mathcal{C}_i$, ω_r denotes the weight of category $c_r \in \widehat{\mathcal{C}}_r$, assigned as shown above. $CatSim(c_r, c_i)$ represents the *category similarity* between c_r and c_i. We consider the two categories c_r and c_i as more similar as the number of items (i.e., recipes) that have been annotated with both the categories increases with respect to the overall number of items annotated with c_r and with c_i. The domain ontology is considered in this case: in fact, given two categories c_i and c_j such that $c_i \sqsubseteq c_j$ (c_i is `subclassOf` c_j), due to the semantics of the `subclassOf` relationship, all recipes annotated with c_i are considered as annotated with c_j as well. For example, $|\texttt{Chicken}| = |\{\texttt{R1}, \texttt{R8}\}| = 2$, $|\texttt{Poultry}| = |\{\texttt{R1}, \texttt{R8}\}| = 2$, $|\texttt{Chicken} \cap \texttt{Poultry}| = |\{\texttt{R1}, \texttt{R8}\}| = 2$, therefore $CatSim(\texttt{chicken}, \texttt{Poultry}) = 1.0$, since $\texttt{Chicken} \sqsubseteq \texttt{Poultry}$. Pairs of categories to be considered in the $Sim_{cat}(\widehat{r}_r, r_i)$ computation are selected according to a maximization function, that relies on the assignment in bipartite graphs and ensures that each category in $\mathcal{C}_i$ participates in at most one pair with one of the categories in $\widehat{\mathcal{C}}_r$ and the pairs are selected in order to maximize the overall $Sim_{cat}(\widehat{r}_r, r_i)$. In the running example, for computing $Sim_{cat}(\widehat{r}_r, \texttt{R1})$, the pair $\langle \texttt{Poultry}, \texttt{Chicken} \rangle$ ($\omega_r = 1.0$) is considered instead of $\langle \texttt{Chicken}, \texttt{Chicken} \rangle$ ($\omega_r = 0.5$) in order to maximize the final result, therefore $Sim_{cat}(\widehat{r}_r, \texttt{R1}) = (1.0 + 1.0 + 1.0)/3 = 1.0$.

Tag-Based Relevance. The relevance of a recipe $r_i = \langle ID_i, n_i, \mathcal{C}_i, \mathcal{T}_i \rangle$ with respect to the request $\widehat{r}_r(u) = \langle \widehat{\mathcal{C}}_r, \widehat{\mathcal{T}}_r, \tau_r \rangle$ taking into account (semantic) tags in $\widehat{\mathcal{T}}_r$ and $\mathcal{T}_i$, denoted with $Sim_{tag}(\widehat{r}_r, r_i) \in [0, 1]$, is computed by evaluating the terminological affinity between pairs of tags, one from the first set ($\widehat{\mathcal{T}}_r$) and one from the second set ($\mathcal{T}_i$), and by combining them through the following formula, that is:

$$Sim_{tag}(\widehat{r}_r, r_i) = \frac{2 \cdot \sum_{t_1 \in \widehat{\mathcal{T}}_r, t_2 \in \mathcal{T}_i} TermAff(t_1, t_2)}{|\mathcal{T}_i|} \in [0, 1] \tag{2}$$

where t_1 and t_2 are tags, $|\mathcal{T}_i|$ denotes the number of items in $\mathcal{T}_i$. The rationale behind Eq. (2) is the same behind $Sim_{cat}()$ computation. The point here is how to compute $TermAff(t_1, t_2) \in [0, 1]$, since t_1 and t_2 might be both semantic and traditional tags. Let's consider t_1 and t_2 as tags semantically disambiguated using WordNet: in this case, the term affinity between t_1 and t_2 is computed as extensively described in [7], where WordNet-based techniques from the literature have been adopted. In all cases where either t_1 and t_2 do not have a disambiguation based on WordNet, we compare the names of terminological items using the normalized Levenshtein distance (thus obtaining a measure $StringSim(\cdot) \in [0, 1]$). In particular, if both t_1 and t_2 have not been disambiguated, then $TermAff(t_1, t_2) = StringSim(t_1, t_2)$. Otherwise, if t_1 has not been disambiguated, while t_2 presents a sense disambiguation (or viceversa), let's denote with $\mathcal{S}_2$ the set of synonyms of t_2, then $TermAff(t_1, t_2) = max_{t_2^i \in \mathcal{S}_2}\{StringSim(t_1, t_2^i)\}$.

The overall feature-based relevance of a recipe r_i with respect to the request $\widehat{r}_r(u)$ is computed as $Sim(\widehat{r}_r, r_i) = \omega_c \cdot Sim_{cat}(\widehat{r}_r, r_i) + \omega_t \cdot Sim_{tag}(\widehat{r}_r, r_i) \in [0, 1]$, where ω_c and $\omega_t \in [0, 1]$ and their sum equals 1.0. The weights ω_c and ω_t are used to balance the two kinds of relevance. In our experiments we considered $\omega_c = 0.5$ and $\omega_t = 0.5$, thus giving the same importance to both the metrics. The recipes included in the set $\mathcal{R}' \subseteq \mathcal{R}$, as output of the *feature-based recipe filtering*, are those whose overall relevance with respect to the request $\widehat{r}_r(u)$ is equal or greater than a threshold $\gamma \in [0, 1]$ set by the user.

Menu Generation and Ranking. Recipes are aggregated into menus that must be compliant with the template τ_r specified in the request $\widehat{r}_r(u)$. This significantly reduces the number of menu configurations to be generated: in fact, a candidate menu can not contain two recipes r_i and r_j annotated with the same `RecipeType`. Generated menus are ranked according to their similarity with: (i) past menu choices made by the user u who is issuing the request for suggestions, represented by the set $\mathcal{M}[u]$; (ii) prescriptions prepared for the user u according to his/her profile, represented by the set $\mathcal{P}[u]$. Since both menus and prescriptions are formally defined as sets of recipes, the building block in this step is the similarity measure between items aggregations (*item aggregation similarity*), that is computed as follows:

$$Sim_{agg}(a_i, a_j) = \frac{2 \cdot \sum_{r_i, r_j} Sim(r_i, r_j)}{|a_i| + |a_j|} \in [0, 1] \tag{3}$$

where a_i and a_j represent the two compared aggregations, r_i (resp., r_j) is an item (i.e., a recipe) included within a_i (resp., within a_j), $|a_i|$ (resp., $|a_j|$) denotes the number of recipes included within a_i (resp., within a_j). Therefore, we consider two aggregations as more similar as the number of similar items in the two aggregations increases.

The final ranking of a generated menu $a_k \in A^*$, recommended to the user u who issued a request for suggestions, is performed through a ranking function $\rho : A^* \mapsto [0, 1]$, computed as follows:

$$\rho(a_i) = \omega_m \cdot \frac{\sum_{a[u] \in \mathcal{M}[u]} Sim_{agg}(a_i, a[u])}{|\mathcal{M}[u]|} + \omega_s \cdot \frac{\sum_{\widehat{a}[u] \in \mathcal{P}[u]} Sim_{agg}(a_i, \widehat{a}[u])}{|\mathcal{P}[u]|} \tag{4}$$

where $\omega_m, \omega_p \in [0, 1]$, $\omega_m + \omega_p = 1.0$, are weights used to balance the impact of past menu choices and prescriptions on the ranking of recommended menus. We have chosen $\omega_m < \omega_p$ (i.e., $\omega_m \cong 0.4$ and $\omega_p \cong 0.6$) in order to stimulate users on improving their food and nutrition habits, without recommending menus and recipes that are too much distant from users' preferences. This is the most innovative aspect of the approach presented here, compared to recent food recommendation literature.

4 Implementation and Experimental Issues

We implemented the food recommendation approach as a web application called PREFer (**P**rescriptions for **RE**commending **F**ood). The *PREFer Web Interface* guides the user through the registration process, the menu recommendation, the publication of new recipes, also supporting semantic disambiguation of tags (through a WordNet-based *Sense Disambiguation module*), both during the publication of new recipes and the formulation of a request for suggestions, using a wizard similar to the one described in [7]. Registration is performed by answering a food frequency survey (FFQ), that is used to collect information about the users in order to compute their BMI and identify their phenotypes [8], to prepare suggested prescriptions. FFQ is composed of a set of questions (whose structure is shown in table below), aiming at identifying the frequency and quantity of assumption for 145 different types of snacks, meat, fish, pasta, soups, products derived from milk, vegetables and fruits, desserts, drinks.

Food category - Snacks, meat, fish, pasta, soups, products derived from milk, vegetables and fruits, desserts, drinks		
Food type	Frequency of assumption	Quantity
(e.g., hamburger)	*(never), (once per month),* *(2–3 times per month), (once per week),* *(twice per week), (3–4 times per week),* *(5–6 times per week), (once per day),* *(2 or more times per day)*	*(small portion)* *(medium portion)* *(big portion)*

Phenotype identification is executed by medical doctors, who participate to the regional project where PREFer is being developed. The description of this task is out of the scope of this paper. To just give an idea, medical doctors are supported in the identification of phenotypes and have a simple web interface at their disposal to prepare and insert prescriptions as sets of recipes, depending on the result of phenotype identification. Prescriptions preparation for a given phenotype is manually performed offline, but prescriptions are automatically assigned to all users classified in the phenotype.

Experiments on our food recommendation approach are being carried to demonstrate the performances of the approach, in terms of average precision of the recommendations, and to verify the impact of the approach in improving the users' habits concerning food and nutrition. Performance tests are being performed on a dataset obtained by extending an existing one[4], containing about 220 k recipes, randomly aggregated into about 100 k menus, where the PREFer system is presenting comparable average precision with respect to recent approaches. To verify the impact of the approach in improving the users' habits, further experiments are being performed on a population of about two hundreds students, equally distributed among males and females, with an age included between 18 and 24. The compliance of users' choices with reference prescriptions, in order to quantify how much the system is able to improve their behaviour, is quantified through the average aggregation similarity between users' choices and reference prescriptions, starting from Eq. (3). Experiments will be carried

[4] http://mslab.csie.ntu.edu.tw/~tim/recipe.zip.

on until September 2015. Monthly, statistics are generated that, with respect to users' profiles, show the percentage of requests and menu choices that are compliant or closer to reference prescriptions. Experiments carried on the first months showed a satisfying increment of closeness between past preferences and reference prescriptions (around 24 % on average, but reaching about 43 % if we consider only users with preferences that are far from the advisable ones, that is, average closeness that is lower than 0.5).

5 Conclusions

In this paper, we presented a menu generation system that uses a recipe dataset and annotations to discover similarity between kinds of food and user's preferences. The system is being developed and tested within a food recommendation regional project, Smart BREAK. This paper has been meant as a complementary approach to recent food recommendation efforts, in order to take into account reference prescriptions schemes for food recommendation that aim at improving users' nutritional habits. In this sense, our approach can be considered as a step forward compared to existing food recommendation proposals, that could be integrated with our system as well. Experimentation is being performed on the approach, but further experiments will be carried on till the end of the Smart BREAK project, in order to check how much the proposed approach is able to effectively improve nutritional habits and lifestyles.

References

1. Gauch, S., Speretta, M., Chandramouli, A., Micarelli, A.: User profiles for personalized information access. In: Brusilovsky, P., Kobsa, A., Nejdl, W. (eds.) Adaptive Web 2007. LNCS, vol. 4321, pp. 54–89. Springer, Heidelberg (2007)
2. Al Nazer, A., Helmy, T., Al Mulhem, M.: User's profile ontology-based semantic framework for personalized food and nutrition recommendation. Procedia Comput. Sci. **32**, 101–108 (2014)
3. Wang, Y., Liu, Z.: Personalized health information retrieval system. In: AMIA Annual Symposium Proceedings, p. 1149 (2005)
4. Abidi, S., Chen, H.: Adaptable personalized care planning via a semantic web framework. In: 20th International Conference of the European Federation for Medical Informatics (2006)
5. Dominguez, D., Grasso, F., Miller, T., Serafin, R.: PIPS: an integrated environment for health care delivery and healthy lifestyle support. In: 4th Workshop on Agent applied in Healhcare ECAI2006 (2006)
6. Suominen, O., Hyvonen, E., Viljanen, K., Hukka, E.: HealthFinland - a national semantic publishing network and portal for health information. Web Semant. Sci. Serv. Agents World Wide Web **7**(4), 287–297 (2009)
7. Bianchini, D., De Antonellis, V., Melchiori, M.: A multi-perspective framework for web API search in enterprise mashup design. In: Salinesi, C., Norrie, M.C., Pastor, Ó. (eds.) CAiSE 2013. LNCS, vol. 7908, pp. 353–368. Springer, Heidelberg (2013)
8. Rankinen, T., Bouchard, C.: Genetics of food intake and eating behavior phenotypes in humans. Ann. Rev. Nutr. **26**, 413–434 (2006)