# Intelligent Food Planning: Personalized Recipe Recommendation

*Jill Freyne and Shlomo Berkovsky*
CSIRO Tasmanian ICT Centre
Hobart, Australia
firstname.lastname@csiro.au

## ABSTRACT
As the obesity epidemic takes hold, many medical professionals are referring users to online systems aimed at educating and persuading users to alter their lifestyle. The challenge for many of these systems is to increase initial adoption and sustain participation for sufficient time to have real impact on the life of their users. In this work we present some preliminary investigation into the design of a recipe recommender, aimed at educating and sustaining user participation, which makes tailored recommendations of healthy recipes. We concentrate on the two initial dimensions of food recommendations: data capture and food-recipe relationships and present a study into the suitability of varying recommender algorithms for the recommendation of recipes.

## Author Keywords
Recommender systems, personalization, food, recipe, collaborative filtering

## ACM Classification Keywords
H.5.m Information Interfaces and Presentation: Miscellaneous

## General Terms
Algorithms, Performance

## INTRODUCTION
With over 1.6 billion adults worldwide classified as obese [8], health care professionals are investigating the use of online systems to influence the general public to change their attitude and behaviour toward a healthy lifestyle. Weight loss systems have progressed from paper recording of diet and exercise to online systems, in which informative content and intelligent services are used to persuade users to alter their behaviour. In these systems users often provide explicit reporting on diet and exercise and browse health related content such as recipes, exercise plans and others. Thus, there is a huge scope for rich user modelling and personalized content delivery services to both sustain user participation with the system and influence their behaviour.

When adopting a healthy lifestyle, many users lack the skills and knowledge required to affect change. With the aid of personalized recommendations we aim to equip users with relevant information to adopt and sustain a healthier lifestyle. One such personalized service, ideally suited to informing diet and lifestyle, is a personalized meal planner. This planner could exploit explicit food preferences, food diary entries, and user browsing behaviour, as well as various other sources, to inform its recipe recommendations.

The domain of food is varied and complex and presents a large challenge to recommendations. To start with, thousands of food items exist, with almost 1000 different vegetables alone. Secondly, food items are rarely eaten in isolation, with a more common consumption tending to be in the combination of dishes. Given the number of food items in existence, the number of possible combinations is exponentially large. Finally, and more complexly, user's opinion on food items can vary quite significantly based on several factors including whether a food item is cooked or raw, if cooked how it is cooked, what quantity of it is included in a recipe and many others. For example, a person may like smoked salmon, but not grilled salmon or may like salmon for dinner, but not for breakfast.

Several recipe recommender systems have been developed in the past. For example, Sobecki et al [9] presented a hybrid recommender using fuzzy reasoning to recommend recipes, Lawrence et al [5] generated recommendations for new food products that might be appealing to supermarket customers, and Svensson et al [10] provided recipe based grocery shopping recommendations. Unlike our work, the above treated the recipe/product as core items and did not break them down into individual food components.

The area of case-based reasoning has seen numerous works in the area of recipe construction from individual ingredients. The Chef [2] and Julia [4] systems both require extensive domain knowledge to create recipes from core ingredients, while the work by Zhang et al [11] exploits existing techniques (e.g., active learning) and knowledge sources (e.g., WordNet) to construct recipes.

The challenges for recipe recommendations are three fold. Firstly, given the number of *recipes* and *foods items*, what


*IUI'10*, February 7–10, 2010, Hong Kong, China.

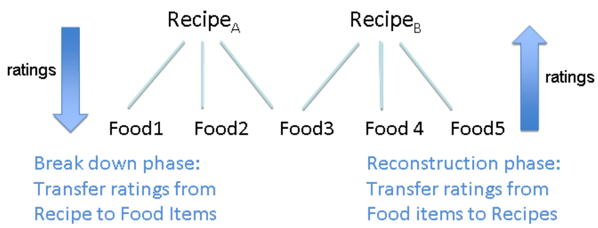


**Figure 1. Menu food relationships**

practical solution exists for gathering sufficient user modelling information on which to base recommendations? In this work we look at ratings on recipes and food items. Secondly, what are the relationships between a recipe and its component foods? If a system is aware of a recipe rating, what assumptions can be made about its ingredients or vice versa. Finally, can preferences on combinations and cooking methods be worked into a recipe recommender?

In this work we concentrate on the first two challenges: data capture and food-recipe relationships. We present a preliminary study into the suitability of varying recommender algorithms for the recommendation of recipes. The study is based on 8701 preferences and ratings provided by 183 users on recipes and food items. We examine the accuracy of content based and collaborative filtering algorithms and compare them with hybrid recommender strategies, which dismantle recipes into their components in order to make accurate recommendations. We show that solicitation of recipe ratings, which are transferred to food ratings through a food's inclusion in recipe, is an accurate and effective method of capturing food preferences.

## RECOMMENDER STRATEGIES

The aim of this work is to uncover which recommender algorithms are suitable for personalized recipe recommendations. We focus on two data gathering strategies: the first is a fine grained food item strategy that gathers explicit ratings on individual food items, the second is a higher level strategy that gathers ratings on recipes. Regardless of whether ratings are gathered on food items or recipes, the output of the recommender algorithms is a *recipe recommendation*.

Before delving into the details of the individual strategies, we explain how we relate foods to recipes and vice versa. In this work, we adopted a simple recipe to food item relationship strategy shown in Figure 1. We ignore all weights, cooking processes and combination effects and consider all food items to be equally weighted within a recipe. Accordingly, we transfer ratings gathered on food items equally to recipes containing these foods and vice versa from recipe ratings to associated foods.

In order to compare our recommender strategies, we implement a baseline algorithm *random* which assigns a randomly generated prediction score to a recipe. The following strategies generate personalized predictions and encompass a pure content based algorithm, a collaborative filtering algorithm and three hybrid algorithms which consist of both content based and collaborative strategies. The strategies are named according to the item types on which their input is based and on the recommender strategies they use. The two strategies which are applied to ratings gathered on *food items* are the $food_{cb}$ and $food_h$ strategy. The $food_{cb}$ strategy assigns scores for a target recipe $r_t$ for a user $u_a$ based on the average of all the ratings provided by $u_a$ on food items $food_1, ..., food_j$ of $r_t$.

$$pred(u_a, r_t) = \frac{\sum_{j \epsilon r_t} rat(u_a, food_j)}{j} \quad (1)$$

The $food_{cb}$ strategy can only make predictions for recipes on which it has information pertaining to the included food items. As mentioned, a huge number of food items exist and gathering a reasonable portion of explicit user preferences on individual food items is unrealistic from a user effort perspective. In order to increase the amount of knowledge held by the system on food items, i.e., reduce the data sparsity, our food hybrid strategy, $food_h$, exploits collaborative filtering to make predictions for unrated food items before carrying out the content matching step. Briefly, a set of $N$ similar users, *neighbours*, is identified using Pearson's correlation algorithm shown in Equation 2 and predictions for food items not rated by $u_a$ are generated using Equation 3.

$$sim(u_a, u_b) = \frac{\sum_{i=1}^{k}(u_{a_i} - \overline{u_a})(u_{b_i} - \overline{u_b})}{\sum_{i=1}^{k}(u_{a_i} - \overline{u_a})^2 \sum_{i=1}^{k}(u_{b_i} - \overline{u_b})^2} \quad (2)$$

$$rat(u_a, food_i) = \frac{\sum_{n \epsilon N} sim(u_a, u_n) rat(u_n, food_i)}{\sum_{n \epsilon N} sim(u_a, u_n)} \quad (3)$$

With this more densely populated food ratings the content-based prediction step from Equation 1 is used to generate a prediction for $r_t$.

In contrast, we also investigate the use of strategies which are applicable to situations were ratings are requested on *recipes* rather than on food items. We implemented four recipe based strategies. The first one, $recipe_{cf}$, is a standard collaborative filtering algorithm assigning predictions to recipes based on the weighted ratings of a set of $k$ neighbours as seen in Equations 2 and 3, where the items in question are recipes rather than food items.

The second, a content based strategy $recipe_{cb}$, breaks down each recipe $r_i$ rated by $u_a$ into food items $food_1, ..., food_x$ (see Figure 1) and assigns the ratings provided by $u_a$ to each food item according to Equation 4.

$$rat(u_a, food_i) = \frac{\sum_{l \text{ s.t. } food_i \epsilon r_l} rat(u_a, r_l)}{l} \quad (4)$$

The strategy then applies the same content based algorithm from Equation 1 to construct a score for the target recipe $r_t$.

We also implemented two recipe hybrid strategies. The first one, $recipe_{hr}$, identifies a set of $N$ neighbours based on *ratings* provided on recipes. Then, it uses Equation 4 to break down each recipes rated by $u_a$ into foods, Equation 3 to predict as many food ratings as possible, and Equation 1 to generate a score for $r_t$. The seconds one, $recipe_{hf}$, differs only from $recipe_{hr}$ in its neighbour selection step. Here user similarity is computed based on overlapping items in the food

Table 1. Rating spread

| hate | dislike | neutral | like | love |
|---|---|---|---|---|
| 1935 | 2300 | 2145 | 1037 | 384 |

matrix established after the recipe break down step rather than on the recipe ratings as in $recipe_{hr}$.

## EVALUATION

In order to test the above recommendation strategies, we gathered user preferences from a set of users, who were taking part in a study on healthy living social technologies [1].

### Set-up

The corpus of recipes used was sourced from the CSIRO Total Wellbeing Diet Book [7]. We extracted 136 recipes which were categorised into *breakfast & lunch*, *soups & salads*, *seafood*, *chicken & pork*, *beef & veal*, *lamb*, and *vegetables*. This set corresponded to 337 food items which were in turn classified into *breads & cereals*, *dairy*, *vegetables*, *general grocery*, and *meat & fish*. On average, each recipe was made up of 9.98 food items and the average number of recipes that each food item was found in was 4.12.

We gathered opinions of 183 users regarding the available recipes and foods over a period of 3 weeks. Users were asked to provide initial preferences on either 20 recipes or 30 foods with the type requested determined randomly. All ratings were captured on a 5-Likert scale, spanning from "Hate" to "Love". In order to obtain a broad spread of items for each user, preferences were requested from randomly selected items in each of the above mentioned recipe or food item categories. Once an initial preference set was gathered, users were asked to periodically provide ratings on additional 20 recipes or 30 foods, depending on their seed set type. In total 8701 preferences were gathered with an average of 47.54 per user. Table 1 shows the spread of ratings over the dataset.

### Methodology

We conducted a traditional leave one out off-line analysis, which took each rating of a $\{user_i,\ r_t,\ rating\}$ tuple from a user profile and used a set of recommender algorithms to generate predictions. For users who provided ratings on food items, the appropriate strategies were $food_{cb}$, $food_h$ and $random$. For users who provided ratings on recipes, the $recipe_{cb}$, $recipe_{cf}$, $recipe_{hr}$, $recipe_{hf}$ and $random$ strategies were used. The accuracy of the recommendations has been evaluated using the MAE measure[1] [3].

### Results

Figure 2 shows the normalized MAE for each strategy. As expected, the $random$ algorithm performed worst with an MAE of 3.88. The poorest performer of the personalized strategies was the collaborative filtering algorithm $recipe_{cf}$.

[1]The aim of the live-user trial was not to judge the accuracy of the recommendations, but to gather preferences to run off-line analysis and bootstrap future studies.

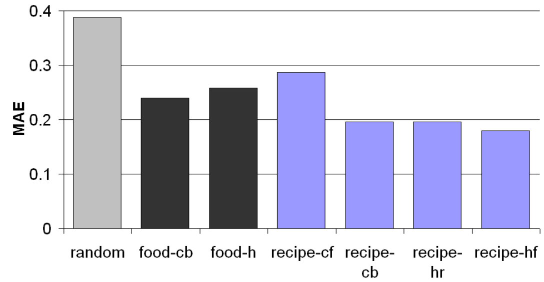


Figure 2. Mean Average Error

This is not surprising given that collaborative filtering algorithms suffer from the cold start problem and the recipe matrix was only 17% populated. The strategies whose input ratings were food based, $food_{cb}$ and $food_h$, fare similarly with an MAE of 0.24 and 0.26, respectively. We note here that the $food_h$ strategy introduced a level of noise to the food rating matrix resulting in a slightly (but significantly) higher MAE than the pure content based strategy $food_{cb}$. This finding is in line with that of Melville et al, who also used content boosted collaborative filtering [6].

The three hybrid recipe based strategies return the lowest MAE of below 0.2 with the $recipe_{hf}$ strategy the most accurate. We do not see an equivalent decrease in accuracy when we compare the $recipe_{cb}$ and $recipe_{hr}$ strategies as we see with the analogous food based strategies $food_{cb}$ and $food_h$. We do however see a difference in accuracy between the two hybrid strategies, $recipe_{hr}$ and $recipe_{hf}$, which differ in the neighbour determination timing. It seems that more accurate neighbour determination and food prediction occurs when neighbours are based on *implied* user ratings on foods rather than on recipes. The differences in MAE are significant at $p < 0.05$ across all pairings except the $recipe_{cb}$ and $recipe_{hr}$ pairs.

The closeness the recipe strategies $recipe_{cb}$, $recipe_{hr}$ and $recipe_{hf}$ is in part due to the coverage of foods in the food matrix. In 7% of simulations the food matrix contained ratings for all the foods in target recipe, such that all three strategies returned the same prediction. Similarly, in 50% of simulations the collaborative filtering was unable to make predictions on any additional food items in the target recipe, again returning the same prediction in all three strategies.

Comparing strategies varying only on the ratings matrix used shows that recommendations made on transferred food ratings outperformed the actual food rating algorithms across the board. The $recipe_{cb}$ strategy has a relative 25% improvement in accuracy over the $food_{cb}$ strategy and similarly the $recipe_{hr}$ and $recipe_{hf}$ strategies have a 25-28% improvement over the $food_h$ strategy. This shows that the decomposition of recipes into food items is beneficial for the purposes of recommendation generation even with a naive break down and reconstruction applied. Also, this shows that ratings on individual food items are not necessarily required for recipe recommendations even if the reasoning occurs on them.

The second aspect pertaining to the practicality of data gathering for food recommendations is algorithm coverage [3].

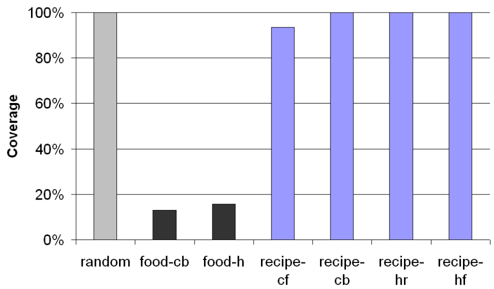


**Figure 3. Coverage**

We wish to find an algorithm which has reasonable accuracy and coverage across all users even for low number of ratings. Figure 3 shows the percentage of simulations where each algorithm was able to generate a prediction. The notable outliers are $food_{cb}$ and $food_h$ algorithms, which have very low coverage of 13% and 16%, respectively. Apart from the $recipe_{cf}$ algorithm, which has a 96% coverage, all other algorithms can generate predictions for all user item pairs.

## CONCLUSIONS AND FUTURE WORK

As with all recommender technologies, a balance needs to be struck between accuracy, coverage and the workload of the users in providing information. This work presented an initial analysis as to the practicality of gathering food preferences and making recipe recommendations. We found that high coverage and reasonable accuracy can be achieved through content based strategies with a simple break down and construction used to relate recipes and food items. We found only marginal improvement in accuracy when collaborative filtering is employed to boost the rating matrix density. In conclusion, we have shown that even a naive recipe break down into food items with reasoning on the latter provides more accurate recommendations than a collaborative filtering algorithm using a sparse matrix.

Our future work includes an investigation into more intelligent means of reasoning on food ratings when recipe ratings are known, and vice versa, on recipe ratings when food ratings are known. Our first consideration concerns the impact of mixed ratings on recipes. For example, when breaking down recipes, a food item may receive a positive rating in one recipe and a negative in another, which are currently just averaged. However, more appropriate combinations would consider whether an ingredient in a positively and a negatively recipe should maintain only the positive rating, as it is unlikely to be the cause of the dislike in the negatively rated recipe. Furthermore, here we operate a simplistic idea of what a recipe recommender needs to do. We are, however, aware that making recipe recommendations is a far more complicated task in reality and we aim to investigate group recommendations, sequential recommendations, and diversification of recommendations.

## ACKNOWLEDGMENTS

This research is jointly funded by the Australian Government through the Intelligent Island Program and CSIRO. The Intelligent Island Program is administered by the Tasmanian Department of Economic Development, Tourism and the Arts. The authors thank Nilufar Baghaei, Stephen Kimani, Dipak Bhandari and Greg Smith for their help with the development of the experimental eHealth family portal.

## REFERENCES

1. N. Baghaei, J. Freyne, S. Kimani, G. Smith, S. Berkovsky, D. Bhandari, N. Colineau, and C. Paris. SOFA: An Online Social Network for Engaging and Motivating Families to Adopt a Healthy Lifestyle. In *Proceedings of 21st Annual OzChi Conference*, pages 269–272, 2009.

2. K. Hammond. CHEF: A Model of Case-Based Planning. In *Proceedings of the Fifth National Conference on Artificial Intelligence*, volume 1, 1986.

3. J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst.*, 22(1):5–53, 2004.

4. T. Hinrichs. Strategies for adaptation and recovery in a design problem solver. In *Proceedings Second Workshop on case-based reasoning, Pensacola Beach, Florida, Morgan-Kauffman*, 1989.

5. R. D. Lawrence, G. S. Almasi, V. Kotlyar, M. S. Viveros, and S. S. Duri. Personalization of supermarket product recommendations. *Data Min. Knowl. Discov.*, 5(1-2):11–32, 2001.

6. P. Melville, R. Mooney, and R. Nagarajan. Content-boosted collaborative filtering for improved recommendations. In *Proceedings of the National Conference on Artificial Intelligence*, pages 187–192. Menlo Park, CA.

7. M. Noakes and P. Clifton. The CSIRO Total Wellbeing Diet Book 2, 2006.

8. W. H. Organization. Chronic disease information sheet. http://www.who.int/mediacentre/factsheets/fs311/en/index.html accessed Sept 2009.

9. J. Sobecki, E. Babiak, and M. Slanina. Application of hybrid recommendation in web-based cooking assistant. In *Proceedings of the Tenth Conference on Knowledge-Based Intelligent Information and Engineering Systems*, pages 797–804, 2006.

10. M. Svensson, J. Laaksolahti, K. Höök, and A. Waern. A recipe based on-line food store. In *IUI '00: Proceedings of the 5th international conference on Intelligent user interfaces*, pages 260–263, 2000.

11. Q. Zhang, R. Hu, B. Namee, and S. Delany. Back to the future: Knowledge light case base cookery. Technical report, Technical report, Dublin Institute of Technology, 2008.