Recipe Recommendation: Accuracy and Reasoning*

Jill Freyne, Shlomo Berkovsky, and Gregory Smith

Tasmanian ICT Center, CSIRO
GPO Box 1538, Hobart, 7001, Australia
firstname.lastname@csiro.au

Abstract. Food and diet are complex domains for recommender technology, but the need for systems that assist users in embarking on and engaging with healthy living programs has never been more real. One key to sustaining long term engagement with eHealth services is the provision of tools, which assist and train users in planning correctly around the areas of diet and exercise. These tools require an understanding of user reasoning as well as user needs and are ideal application areas for recommender and personalization technologies. Here, we report on a large scale analysis of real user ratings on a set of recipes in order to judge the applicability and practicality of a number of personalization algorithms. Further to this, we report on apparent user reasoning patterns uncovered in rating data supplied for recipes and suggest ways to exploit this reasoning understanding in the recommendation process.

Keywords: Collaborative filtering, content-based, machine learning, recipes, personalization.

1 Introduction

The World Health Organisation is predicting that the number of obese adults worldwide will reach 2.3 billion by 2015, a statistic which is attracting increased attention [1]. Much of this attention is being paid to online diet monitoring systems, which have been replacing traditional pen-and-paper programs in recent years. These systems, which often include informative content and services to persuade users to alter their behaviour, gather a vast amount of user preference information that could be harnessed to personalize interactive features in order to increase engagement with the online system, and in turn with the diet program. Dieters use these systems to acquire knowledge, to plan and to record dietary intake. A personalized service ideally suited to informing diet and lifestyle is a personalized recipe recommender. This recommender could exploit explicit food

* This research is jointly funded by the Australian Government through the Intelligent Island Program and CSIRO Food and Nutritional Sciences. The Intelligent Island Program is administered by the Tasmanian Department of Economic Development, Tourism, and the Arts. The authors acknowledge Mealopedia.com and Penguin Group (Australia) for permission to use their data.

Joseph A. Konstan et al. (Eds.): UMAP 2011, LNCS 6787, pp. 99–110, 2011.

ratings, food diary entries, and browsing behaviour to inform its recommendations and assist dieters with one of the biggest challenges of successful lifestyle change.

The domain of food is varied and complex and presents many challenges to the recommender systems community. There are many factors that will impact on a user's opinion on foods, some of which will be more important to some individuals than others. The obvious contributory factors are *cooking methods*, *ingredients*, *costs* and *availability*, *cooking complexity*, *preparation time*, *nutritional breakdown*, *ingredient combination effects*, as well as *user goals*, *cultural* and *social factors*. Add to these factors the sheer number of available ingredients, the fact that eating often occurs in groups, that the sequencing is crucial, and the complexity of challenge becomes clear.

In this work, we follow on from earlier preliminary analysis on the suitability of traditional personalization algorithms for recommendations in the food domain. We explore the possibilities of using machine learning and analyse trends in users' reasoning, which uncover user traits that could have significant impact in many dimensions of recommender techniques. Thus, the contributions of this work are (1) an analysis reporting on the applicability of various personalized techniques for rating prediction, and (2) a report on the observed trends of reasoning uncovered by machine learning feature selection algorithm.

The paper is structured as follows; Section 2 positions this work in relation to other work in the field, Section 3 provides details of the recommendation algorithms implemented. In Section 4 we discuss algorithm accuracy and performance and the trends uncovered in the ratings sets of users. We conclude with a discussion of our findings and present an outline of future plans.

2 Related Work

Initial efforts to address the challenge of intelligent support in meal planning resulted in systems, such as Chef [6] and Julia [9], which rely heavily on domain knowledge for recommendations. More recently, works concentrating on social navigation, ingredient representation and recipe modeling have come to the fore. A recipe recommender system based on user browsing patterns is presented by Svensson et al. [14]. They use social navigation techniques and apply collaborative filtering to predict ratings. While users reported liking the system, formal analysis of its predictive power is not reported.

Freyne et al. investigated the performance of collaborative, content-based, and hybrid recommender strategies, which break down recipes into ingredients in order to generate recommendations [2,3]. Their results showed that solicitation of recipe ratings, which are transferred to ingredient ratings, is an accurate and effective method of capturing ingredient preferences, and that the introduction of simple intelligence can improve the accuracy of recommendations.

Zhang et al. also make use of an ingredient representation but, in contrast, distinguish three levels of importance, which are manually assigned [17]. Using this mechanism, ingredients that are considered to be more important have the largest contribution to the similarity score. Once again, a level of domain expertise is required for this process. We would argue that the importance of an ingredient in a recipe is likely to be user dependent rather than a generic rule. Pixteren et al. do take a user-centered approach to recipe modeling rather than making a priori assumptions about the characteristics that determine the perceived similarity, such as ingredients or directions [15]. They derive a measure, which models the perceived similarity between recipes by identifying and extracting important features from the recipe text. Based on these features, a weighted similarity measure between recipes is determined.

3 Recommender Strategies

This work aims to investigate how individuals reason in relation to food and in particular recipes. We examine real user rating data to see if patterns of reasoning exist for individuals. This analysis presented here aims to understand reasoning on recipes only, as a first step, and disregards the context of meal planning and scheduling. We acknowledge that other factors are at play when planning meals but it is crucial to get the foundations right before embarking on a total solution to this complex problem.

Each recipe in our corpus has a basic structure including a *Title*, *Ingredient List* and *Instructions*. From this basic information we automatically extract additional information. We decipher two indicators of recipe complexity: the *number of ingredients* and the *number of steps* required to complete the recipe. In addition, we manually annotate each recipe with simple domain knowledge in the form of a *general cuisine type*, a *specific cuisine type*, and a *broad category*, containing options traditionally used to classify a dish. The options for cuisine types and categories are in Table 1.

We implemented three personalized recommender algorithms: two standard recommender strategies and one machine learning strategy suitable for rating prediction. A standard *collaborative filtering* algorithm [10] assigns predictions, $pred(u_a, r_T)$, for user u_a for a target recipe, r_T, based on the weighted ratings of a set of N neighbours. Briefly, each user's similarity to u_a is determined as shown in Equation 1 and the users with the top N similarity scores make up the neighbours. Predictions for r_T are generated using Equation 2.

Table 1. Metadata features and values

General Cuisine	Specific Cuisine	Category
African, American, Asian, European, International, Oceania	African, Australian, Chinese, Eastern European, French, German, Greek, Indian, International, Italian, Japanese, Mexican, Middle Eastern, South East Asian, Southern, Spanish, UK&Ireland	beef, pork, lamb, chicken, veal, fish, vegetables, fruit

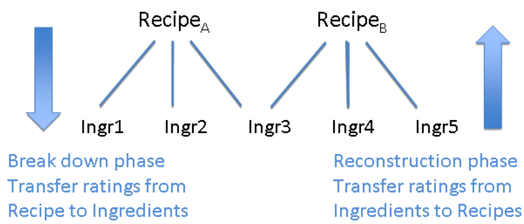

Fig. 1. Recipe - ingredient breakdown and reconstruction

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

$$pred(u_a, r_T) = \frac{\sum_{n \epsilon N} sim(u_a, u_n) rat(u_n, r_T)}{\sum_{n \epsilon N} sim(u_a, u_n)} \quad (2)$$

The second algorithm is a *content-based* algorithm [3], which breaks down each rated recipe into ingredients $ingr_1, ..., ingr_x$ (see Figure 1) and assigns the provided rating to each ingredient according to Equation 3. We transfer the ratings gathered for each recipe to each ingredient listed in the recipe equally. The strategy then applies a content-based algorithm shown in Equation 4 to predict a score for the target recipe based on the average of all the scores provided by the user on ingredients $ingr_1, ..., ingr_j$ making up the target recipe.

$$score(u_a, ingr_i) = \frac{\sum_{r \epsilon recipes(ingr_i)} rat(u_a, r)}{\#recipes(ingr_i)} \quad (3)$$

$$pred(u_a, r_t) = \frac{\sum_{i \epsilon ingredients(r_t)} score(u_a, i)}{\#ingredients(r_t)} \quad (4)$$

Our third algorithm is a sophisticated prediction algorithm using the open source data mining tool Weka [5]. We used the logistical decision tree algorithm M5P [16,13] to predict scores based on the recipe content and metadata. The M5P algorithm can be applied to all or a subset of the recipe features, including the presence and absence of ingredients and the associated metadata.

M5P is a binary tree classifier, where each leaf predicts a numeric quantity using linear regression [13]. Each data instance is a set of features $\{a_1, \ldots, a_{N+1}\}$, where each feature may be numeric or nominal, but a_{N+1} is the class label and must be numeric. Predictions are made based on the smoothed tree by tracing the path to a leaf and using a linear interpolation of predictions made by the nodes on the path. Each non-leaf node performs a binary test of a single feature from $\{a_1, \ldots, a_N\}$, partitioning instances into those to be classified by the right and left sub-tree. Each leaf node is a most specific generalisation that contains a linear regression model, predicting the class label for those instances that are classified by this leaf, such the set of leaves of the tree collectively predicts the class label over the whole space.

Model tree induction by M5P occurs in three stages. In the first stage, nodes are recursively split using a criterion that minimizes the intra-subset variation in the class values down each branch. For each candidate feature to test at that node, the expected reduction in error resulting from testing that feature is computed. A node is split on the best feature if the highest expected reduction in error is large enough. In the second stage, the tree is simplified by pruning. Linear models are computed for non-leaf nodes, starting at the bottom, and error estimates are compared to the corresponding leaf nodes. If the non-leaf is chosen, that sub-tree is pruned and replaces with a new leaf node.

4 Evaluation

We gathered a dataset of recipe ratings through Mechanical Turk, Amazon's online task facilitator (www.mturk.com). A corpus of 343 recipes was obtained from the CSIRO Total Wellbeing Diet books [11,12] and from the meal planning website Mealopedia.com (www.mealopedia.com).

Online surveys, each containing 35 randomly selected recipes, were posted to the system. Responses for each of the 35 recipes displayed were required and users could answer as many of the published surveys as they wished. Each question asked users to report on how much a recipe appealed to them on a 5-Likert scale, spanning from "not at all" to "a lot". Overall, we gathered 101,557 ratings of 917 users, such that the density of the obtained ratings matrix was 33%. 15% (15191) of recipes were rated *not at all*, 14% (14425) – *not really*, 20%(19840) – *neutral*, 25% (25593) – *a little*, and 26% (26508) – *a lot*.

On average, each recipe was made up of 9.52 ingredients (stdev 2.63) and the average number of recipes that each ingredient was found in was 8.03 (stdev 19.8). On average, each user rated 109 recipes (stdev 81.9), with the minimum number of ratings per person being 35 and the maximum being 336.

4.1 Set-up

We conducted a number of experiments on the dataset collected using traditional recommender and machine learning approaches, to determine algorithm accuracy for recipe rating predictions. For the collaborative filtering (CF) and content based (CB) algorithms, we employed a traditional leave-one-out analysis, which removed each tuple $\{u_i,\ r_t,\ rat(u_i, r_t)\}$ from the user's profile and used the algorithms to predict the rating $rat(u_i, r_t)$. A set of 20 neighbours were selected only once for each user, based on the entire set of ratings provided. The M5P algorithm was run separately on the ratings of each user. Each user profile was split into 90% training and 10% test set and the ratings $rat(u_i, r_t)$ in the test set were predicted. 10 iterations were carried out for different selections of the test set. We present the average MAE [8] score obtained by each algorithm.

4.2 Algorithm Accuracy

Table 2 shows the average MAE of the prediction scores for each algorithm presented in section 3. The results of the CF and CB algorithms match earlier

Table 2. MAE of personalized algorithms

Content Based Filtering	Collaborative Filtering	Machine Learning (M5P)
1.2083	1.2614	0.9774

results from a similar analysis on an smaller dataset presented in previous works [3,2]. The accuracy of CF and CB recommenders is similar, with an increase in accuracy of only 0.05 over CF obtained by CB. A comparison between the CF algorithm, which treats each recipe as one entity and ignores its ingredients, and the CB algorithm, which considers the ingredients, shows that even the uniformly weighted break down and reconstruction offer increases in accuracy.

The best performing algorithm is the M5P algorithm, which in this case takes only the recipe metadata into consideration to determine recipe ratings. The M5P algorithm is the most accurate, with an MAE of 0.98. It is worth noting that we also ran this analysis using a linear regression algorithm, but the results were very similar to those of the M5P algorithm and the results provided by the M5P algorithm facilitated a more in-depth analysis of user behaviour, thus we omitted the results and discussion due to space limitations.

In terms of the coverage of each algorithm [8], the *M5P* strategy achieved a 100% coverage for each user, whereas the CB strategy obtained 92% coverage and the CF strategy only 83.8%. Thus, the machine learning approach appears to be the best performer overall.

4.3 Reasoning on User Input

While knowing which algorithm performs best is valuable, we embarked on further investigation into the reasons behind the improved performance of the M5P algorithm. By understanding the differences in performance we can affect other dimensions of recommender systems such as information gathering for user profiling, hybridization of recommendation algorithms, and persuasive explanation of recommendations.

As mentioned, we use three classes of metadata: *complexity data* that details the number of steps and ingredients in a recipe, *cuisine* data that categorises recipes according to their cuisine type, and the *broad category* which categorises recipes according to the main food type included in the recipe.

We employed a Corrleation-based Feature Selection algorithm (CFS) to compute a heuristic measure of the "merit" of feature subsets from pair-wise feature correlations. Merit is quantified by

$$M_S = \frac{k\overline{r_{cS}}}{\sqrt{k + k(k-1)\overline{r_S}}} \tag{5}$$

where k is the number of features in the selected set S, $\overline{r_{cS}}$ is the mean feature-class correlation over class c and set of features S, and $\overline{r_S}$ is the average feature-feature intercorrelation over S. The correlation is calculated using symmetrical uncertainty:

Table 3. Distribution of Predictors

	1 predictor	2 predictors	3 predictors	4 predictors
profiles	172	327	187	147
% of total	20.6%	39.2%	22.4%	17.7%

$$u(X,Y) = 2\left[\frac{g(X,Y)}{h(Y)+h(X)}\right] \tag{6}$$

where h is entropy of a feature and g is information gain of a class given a feature [4]. Thus, selection of a feature as a predictor depends on the extent to which it predicts classes in areas of the instance space not yet predicted by other features.

We analyzed the set of predictive features selected for each user in our dataset. 20.6% of users have one predictive feature, 39.2% have two, 22.4% have three and 17.7% have four predictive features, as seen in Table 3. We hypothesize that the different number of predictors reflects different levels of reasoning employed by users when providing ratings. To ascertain whether the number of predictive features is related to the number of user's ratings, we calculated the correlation between the density of a user's rating vector and the number of features selected. The correlation coefficient was -0.031, showing no patterns between the number of ratings provided by a user and the number of predictive features.

20.6% of users have one predictive feature selected. For 93% of this group, the feature identified was the *broad category* feature, i.e. the presence of a certain key ingredient. We assume that users in this group assign ratings to recipes based primarily on the main ingredient of the recipe. Simple rational following this reasoning is: *"I like chicken recipes, I dislike fish recipes, I love beef recipes, etc"*.

39.2% of users have two predictive features selected and we assume are reasoning on two levels. In 96% of these profiles, the *broad category* feature was selected, this time in conjunction with an additional feature. The additional feature selected was the *general cuisine* feature in 48.6% of cases, the *specific cuisine* in 37.3% of cases, or *number of ingredients* in 10.4% of cases. Table 4 shows how this breaks down for the various combinations of features. The dominance of the *broad category* feature changes depending on its coupling with other features. For example, when coupled with *general cuisine*, the *broad category* feature is the most predictive feature in 57.2% of cases. So, with respect to the *broad category* and *general cuisine* features, 57.2% of users are rationalizing according to the statements *"I like beef and I love it when its included in a Chinese style dish"* and 42.8% of according to *"I love Chinese dishes, especially ones which contain beef"*. When the *specific cuisine* feature is a predictor in conjunction with the *broad category*, in 81.9% of cases the *broad category* is the most predictive feature and only in 18.1% of cases the *specific cuisine* feature is most predictive. The opposite is the case when the *number of ingredients* feature is present. It is the dominating predictive feature in 74.6% of cases, while the *broad category* is the most predictive feature in 26.4% of cases.

Table 4. Combinations and dominance of features when two predictive features exist

Predictive features (feat1,feat2)	% of profiles applicable	most predictive feat1	most predictive feat2
(broad category, general cuisine)	48.62%	57.2%	42.8%
(broad category, specific cuisine)	37.31%	81.9%	18.1%
(broad category, number of ingredients)	10.40%	26.4%	74.6%
other	5.37%		

Table 5. Combinations and dominance of features when three predictive features exist

Predictive features (feat1, feat2, feat3)	% of profiles applicable
(number of ingredients, general cuisine, specific cuisine)	43.28%
(number of ingredients, specific cuisine, broad category)	20.90%
(number of ingredients, general cuisine, broad category)	18.51%
(general cuisine, specific cuisine, broad category)	11.94%
other	5.37%

20% of users have three predictive features selected. When users are reasoning on three features, the *broad category* is *not* a predictive feature in 43.3% of cases. This suggests that when users are applying complex reasoning processes to provide well thought ratings, their focus is on the fine grained details of cuisine type and cooking complexity, rather than simply on the main ingredient of the recipe itself. These users are likely to reason along the lines of *"I like Asian dishes, in particular Thai dishes, but only ones with a small number of ingredients"*. Table 5 shows the break down of the three predictive features.

4.4 Applications of reasoning knowledge

One of the challenges of recommender systems is that of the cold start problem, where insufficient user information has been attained to generate accurate recommendations. One way of combatting this is to gather ratings for items that are seen to attract varied ratings from users (i.e items that some love and others hate, rather than items that most tend to like or dislike). Gathering ratings on these items maximises the information gained from each individual rating [7]. To achieve similar goals, we consider using the feature selection process as an indicator for the number and type of reasoners that a user is using when providing rating recipes. This information would allows us to (1) obtain ratings that provide maximal differentiation across the desired features, and (2) determine how many of these ratings are required for accurate user profiling as well as influencing other areas of the recommendation process.

In the following analysis we concentrate on users with more than 100 ratings in their profiles. For each user, the number of features on which they reason is determined by examining the first 100 ratings provided. In this experiment, we

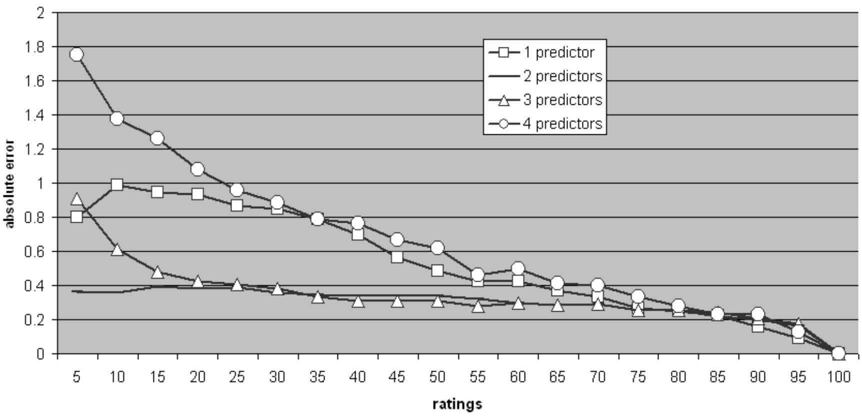

Fig. 2. Predictor stability over time

grow the number of ratings in the profile, k, from 5 to 100 in randomly selected increments of 5 ratings. For each k we carry out the feature selection process and compare the number of selected features to the number of features selected when all 100 ratings in the profile are considered. We repeat this process 10 times and report on the average error between the two. We compute the error separately for groups of users reasoning on 1, 2, 3, and 4 features. Figure 2 shows the average error for various values of k.

The highest error is obtained for users reasoning on 4 features. We observe an error rate of 1.75 for $k = 5$, an initial steep drop off followed by a steady decline. The same trend is seen for users reasoning on 3 features, although the error at $k = 5$ is half that of the previous group. This curve levels off at 0.4 when $k = 25$. A very consistent error line is observed for users reasoning on 2 features, showing that the feature selection is accurate even when a small number of ratings is available. In contrast to the emerging trend, the error rates are high for users reasoning on only 1 feature. The error hovers around the $0.8, ..., 1$ mark until 35 ratings are received and then steadily decreases. Note that when a user is reasoning on 1 feature, the error can only be positive (i.e., the algorithm selected multiple features), whereas in other cases it could over or under predict. Thus, the feature selection is mostly predicting that the users are reasoning on two features rather than one for $k < 35$. We believe that this is caused by the lack of dominance of the main feature, when insufficient ratings are available for the feature's merit score to be sufficient independent.

Figure 3 shows the MAE of predictions made using the selected features for user profiles of different sizes of $5, .., 100$. For each value of k, feature selection was completed on 90% of the user profile and the selected features used to predict the remaining 10%. 10 runs of each were carried out and the average MAE across users in each group are reported. Note that a similar accuracy is obtained for users reasoning on two and three features when $k > 5$. However, there is a distinct difference in the accuracy of predictions for users reasoning

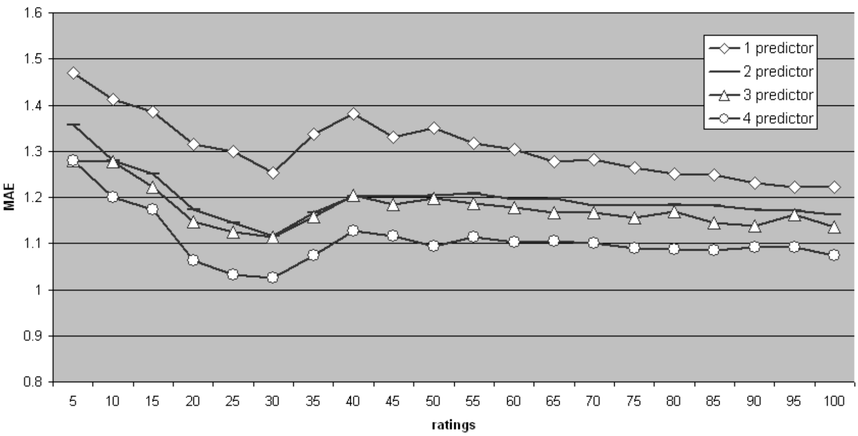

Fig. 3. MAE of predictions made using feature selection at various k

on one feature and four features. These groups had very similar absolute errors in Figure 2, but the error has affected the prediction accuracy in different ways. The average error observed for the number of selected features (Figure 2) across users reasoning on one feature at $k = 10$ was 1.0. This error however, was always a positive error and the number of selected features was being over predicted, resulting in an additional layer being added to the decision tree. Similarly, at $k = 10$ the average error for users reasoning on four features was 1.4, and this was always a negative error. Hence, the number of selected features was under predicted and a shallower tree, of on average 2.6 levels, was generated. So, in the overestimated cases, noise was added to the tree, and in the underestimated cases the tree was incomplete.

We examined the changes of merit scores when additional noisy data is added to a tree and when information is missing from a tree. The analysis shows a 10% reduction in merit score when an additional feature was selected. Thus, the correlation between the features in the tree and the ratings is 10% lower. However, missing information from the tree has a weaker effect. In this case, the information loss associated with one missing feature is 2% and with two missing features is 4%. Thus, it appears better to underestimate the number of predictors rather than overestimate them. Hence, the MAE scores obtained for users with four predictive features are lower than those obtained for users with one predictive feature.

4.5 Summary

The results of this exploratory work have uncovered several useful and informative trends in how users approach a recipe rating task and on which domain features they are reasoning. We uncovered four groups of users, each reasoning on recipes on different levels. The first group, which consisted of 20.6% of users, reasoned on the most general metadata – the *broad category* of the recipes. The largest group of users (39.2%) reasoned on two features, and in most cases the features in question related to the *broad category* and *cuisine type* metadata, showing a deeper reasoning process. When users reasoned on three features (22.4%), they often did not reason on the *broad category* but preferred other more detailed features such as recipe complexity measures as well as cuisine types. Finally, 17.4% of users reasoned on four features.

Further analysis showed that it is easy to select the features, on which users reasoning on two features actually reason, even with few ratings. On the contrary, selecting features that users reasoning on one feature and on four features requires more ratings. Finally, we noted that the accuracy of the feature selection technique has different effects depending on whether too many or two few features are selected. This is explained by the decision tree based prediction mechanism employed by M5P.

5 Conclusions and Future Work

In this work we have investigated the applicability of recommender techniques to generate recipe recommendations and identified the performance enhancements achieved by using machine learning techniques. Analyses of the results have shown that users reason on various levels when rating recipes and that various combinations of metadata are seen to have different predictive qualities for different users. This information assists us in understanding how users provide recipe ratings and suggests ways in which this knowledge could be used to benefit recommender algorithms.

As mentioned, implications of knowing how users reason are obvious in the recommender domain. Informative rating acquisition is a logical next step for evaluation. We will develop an active learning model, which will determine a user's reasoning level and adapt the ratings requested accordingly, in order to obtain the most high value user information. Item diversity is another example of where knowing the reasoning process is important, particularly when sequencing recommendations as in the food domain. Recipe diversity could depend on the user, rather than just on the recipe similarity. In a similar way, persuasive techniques aiming to increase the uptake of recommendations could be made more effective, if the user's reasoning process is known.

References

1. Chronic disease information sheet, `http://www.who.int/mediacentre/factsheets/fs311/en/index.html` (accessed June 2010)
2. Freyne, J., Berkovsky, S.: Intelligent Food Planning: Personalized Recipe Recommendation. In: Proceedings of the 2010 International Conference on Intelligent User Interfaces (IUI 2010), pp. 321–324 (2010)
3. Freyne, J., Berkovsky, S.: Recommending Food: Reasoning on Recipes and Ingredients. In: De Bra, P., Kobsa, A., Chin, D. (eds.) UMAP 2010. LNCS, vol. 6075, pp. 381–386. Springer, Heidelberg (2010)

4. Hall, M.: Correlation-based feature selection for machine learning. PhD thesis, Citeseer (1999)
5. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The Weka Data Mining Software: An Update. ACM SIGKDD Explorations Newsletter 11(1), 10–18 (2009)
6. Hammond, K.: CHEF: A Model of Case-Based Planning. In: Proceedings of the Fifth National Conference on Artificial Intelligence, vol. 1 (1986)
7. Herlocker, J.L., Konstan, J.A., Borchers, A., Riedl, J.: An algorithmic framework for performing collaborative filtering. In: SIGIR, pp. 230–237 (1999)
8. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. 22(1), 5–53 (2004)
9. Hinrichs, T.R.: Strategies for adaptation and recovery in a design problem solver. In: Proceedings of the Workshop on Case-Based Reasoning (1989)
10. Konstan, J.A., Miller, B.N., Maltz, D., Herlocker, J.L., Gordon, L.R., Riedl, J.: GroupLens: applying collaborative filtering to Usenet news. Communications of the ACM 40(3), 87 (1997)
11. Noakes, M., Clifton, P.: The CSIRO Total Wellbeing Diet Book. Penguin Group, Australia (2005)
12. Noakes, M., Clifton, P.: The CSIRO Total Wellbeing Diet Book 2. Penguin Group, Australia (2006)
13. Quinlan, J.: Learning with continuous classes. In: Proceedings of the 5th Australian Joint Conference on Artificial Intelligence, pp. 343–348. Citeseer (1992)
14. Svensson, M., Höök, K., Laaksolahti, J., Waern, A.: Social navigation of food recipes. In: CHI 2001: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 341–348. ACM, New York (2001)
15. van Pinxteren, Y., Geleijnse, G., Kamsteeg, P.: Deriving a recipe similarity measure for recommending healthful meals. In: Proceedings of the 2011 International Conference on Intelligent User Interfaces, IUI 2011, pp. 105–114 (2011)
16. Wang, Y., Witten, I.: Induction of model trees for predicting continuous classes (1996)
17. Zhang, Q., Hu, R., Namee, B., Delany, S.: Back to the future: Knowledge light case base cookery. Technical report, Dublin Institute of Technology (2008)