Group-Based Recipe Recommendations: Analysis of Data Aggregation Strategies

Shlomo Berkovsky and Jill Freyne
CSIRO Tasmanian ICT Center
GPO Box 1538, Hobart, 7001, Australia
firstname.lastname@csiro.au

ABSTRACT

Collaborative filtering recommendations were designed primarily for individual user models and recommendations. However, nowadays more and more scenarios evolve, in which the recommended items are consumed by groups of users rather than by individuals. This raises the need to uncover the most appropriate group-based collaborative filtering recommendation strategy. In this work we investigate the use of aggregated group data in collaborative filtering recipe recommendations. We present results of a study that exploits recipe ratings provided by families of users, in order to evaluate the accuracy of several group recommendation strategies and weighting models, and analyze the impact of switching strategies, data aggregation heuristics, and group characteristics on the performance of recommendations.

Categories and Subject Descriptors

H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces

General Terms

Algorithms, Design, Experimentation

Keywords

Recipe recommendations, group recommendations, collaborative filtering

1. INTRODUCTION

The vast amount of digital information highlights the emergent need for personalized recommendation tools, which help users to identify the most relevant items. Collaborative filtering (CF) is one of the most widely-used statistical recommendation techniques [10, 3]. It can predict the interest level of a user for a previously unrated item by aggregating opinions of similar users, who have already rated this item. Generally, CF is designed to aggregate opinions of individual users and produce individual recommendations.

RecSys2010, September 26–30, 2010, Barcelona, Spain.

However, as the use of recommender systems increases, we face more and more scenarios and application domains, in which the recommended items are inherently consumed by groups of users rather than by individuals. Consider music selection in public places [12], tourist attractions [1], holiday destinations [13], movies [15], and TV programs [16], as examples of recommendations more suited to groups than to individuals. In these scenarios, the recommendations should be tailored to the entire group, to ensure maximum satisfaction of each member and the group as a whole.

To implement group recommendations using CF, it was proposed to aggregate the individual user data of the group members (either preferences or recommendations) into group-based data, and then use the aggregated data in the CF recommendation process [9]. Although group recommendations are not as accurate as personalized ones, they have the potential to be more accurate than the general recommendations, which are the natural fall back when personalized recommendations are not achievable.

In this work we investigate the applicability of CF family-based recipe recommendations, a particular case of group recommendations, for the purpose of uncovering which strategy is most appropriate when generating CF recommendations for a group. Recipe and food consumption are good examples of a group activity, as typically all family members eat a joint meal at least once a day. Hence, a system providing recipe recommendations for a family should consider the preferences of all family members, satisfy each member to the maximal extent, while not recommending a recipe that will be completely rejected by a member.

We implemented four strategies and four weighting models for aggregating individual data into family-based data. We evaluated CF recommendations generated using the aggregated data against real-life recipe ratings, provided by families interacting with experimental eHealth portal. The results showed that the most appropriate family-based recipe recommendation strategy should (1) aggregate individual user models rather than individual recommendations, and (2) weight individual users according to their observed activity rather than according to pre-defined assumptions. We also analyzed the impact of switching strategies [5], data aggregation heuristics [11], and group characteristics on the performance of the generated CF recommendations.

Hence, the contributions of this work are two-fold. Firstly, we evaluate the performance of several CF group recommendation strategies and weighting models and uncover the most appropriate group-based strategy. Secondly, we analyze the impact of switching strategies, data aggregation heuristics,

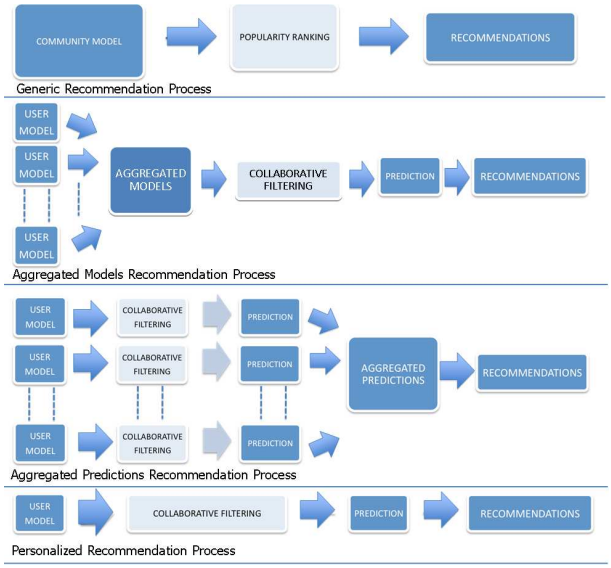

Figure 1: Recommendation generation process

and group characteristics on the performance of recommendations.

This paper is structured as follows. Section 2 overviews related group-based recommendation research. Section 3 presents the developed strategies and models. Section 4 discusses the experimental results. Finally, Section 5 concludes the paper and outlines future research directions.

2. RELATED WORK

Due to the large number of activities, which users carry out as part of a group rather than individually, recommender systems research has embraced the topic of group-based recommendations [9]. Group-based recommendations are pertinent to many domains and applications, such as music [12], movies or TV programs [15, 16], tourism [1, 13], online communities [6], and others.

To date, group recommendations have been mostly generated using two strategies: aggregating individual models into group models (*aggregated models*) or aggregating individual predictions into group predictions (*aggregated predictions*). These strategies differ in the timing of data aggregation step, as depicted in Figure 1. The *aggregated models* strategy [7] merges individual user models into a group-based model and then generates recommendations using the aggregated group model. The *aggregated predictions* strategy [11] generated individual predictions and then aggregates the individual predictions into a group prediction.

As highlighted in [9], the selection between the *aggregated models* and the *aggregated predictions* strategies often depends on external considerations, such as the ability to negotiate group preferences, priorities and social dynamics, privacy constraints, and ability to explain the recommendations. However, in many scenarios either strategy is applicable and, to the best of our knowledge, no prior research compares the recommendations generated using the two strategies. Hence, this work focuses on empirical evaluation and comparison of group-based recommendation strategies using a dataset of recipe ratings of real families of users.

3. GROUP BASED RECOMMENDATIONS

The primary aim of this work is to uncover which strategy for data aggregation data and group recommendation generation is most appropriate when dealing with groups that are made up of users within a nuclear family structure.

3.1 Recommendation Strategies

We compare the four recommendation strategies shown in Figure 1. The *general* strategy exploits the wisdom of the crowd and recommends the most popular items. The *aggregated models* and *aggregated predictions* strategies exploit the two group-based recommendation algorithms. Finally, the *personalized* strategy exploits a standard CF algorithm.

The *general* strategy recommends most popular, *i.e.*, most highly rated, items to users [4]. Each unrated item $item_i$ is assigned a prediction score $pred(item_i)$ based on the ratings $rat(u_x, item_i)$ of n users in $u_x \in U$, who rated $item_i$, as shown in equation (1).

$$pred(item_i) = \frac{\sum_{x \in U} rat(u_x, item_i)}{n} \quad (1)$$

The group-based *aggregated models* strategy [2] initially computes a family rating $rat(f_a, item_i)$ for family f_a and $item_i$ by aggregating the individual ratings $rat(u_x, item_i)$ of family members $u_x \in f_a$, who rated $item_i$, according to their relative weight $\omega(u_x, f_a)$, as shown in equation (2).

$$rat(f_a, item_i) = \frac{\sum_{x \in f_a} \omega(u_x, f_a)\, rat(u_x, item_i)}{\sum_{x \in f_a} \omega(u_x, f_a)} \quad (2)$$

Then, CF is applied to the family model, as shown in equation (3). A prediction $pred(f_a, item_i)$ for family f_a and unrated item $item_i$ is generated by computing similarity degree $sim(f_a, f_b)$ between f_a and all other families $f_b \in F$ and aggregating family ratings $rat(f_b, item_i)$ of families, which rated $item_i$, according to the similarity degree $sim(f_a, f_b)$.

$$pred(f_a, item_i) = \frac{\sum_{f_b \in F} sim(f_a, f_b)\, rat(f_b, p_i)}{\sum_{f_b \in F} sim(f_a, f_b)} \quad (3)$$

Finally, $pred(f_a, item_i)$ is assigned to all family members, *i.e.*, $pred(u_x, item_i \mid u_x \in f_a) = pred(f_a, item_i)$.

The group-based *aggregated predictions* strategy [2] initially generates individual prediction $pred(u_x, item_i)$ for user u_x and unrated item $item_i$ using the standard CF algorithm, as shown in equation (4). The prediction is generated by computing the degree of similarity $sim(u_x, u_y)$ between the target user u_x and all other users $u_y \in U$ and aggregating individual ratings $rat(u_y, item_i)$ of users, who rated $item_i$, according to the similarity degree $sim(u_x, u_y)$.

$$pred(u_x, item_i) = \frac{\sum_{y \in U} sim(u_x, u_y)\, rat(u_y, item_i)}{\sum_{y \in U} sim(u_x, u_y)} \quad (4)$$

Then, the process becomes group-focused. To generate prediction $pred(f_a, item_i)$ for family f_a and item $item_i$, individual predictions $pred(u_x, item_i)$ of family members $u_x \in f_a$ are aggregated according to their relative weight $\omega(u_x, f_a)$, as shown in equation (5).

$$pred(f_a, item_i) = \frac{\sum_{x \in f_a} \omega(u_x, f_a)\, pred(u_x, item_i)}{\sum_{x \in f_a} \omega(u_x, f_a)} \quad (5)$$

Finally, $pred(f_a, item_i)$ is assigned to all family members, *i.e.*, $pred(u_x, item_i \mid u_x \in f_a) = pred(f_a, item_i)$.

The *personalized* strategy examines users individually, users regardless of their family membership using the standard CF algorithm [10]. For each user u_x, unrated item $item_i$ is assigned a predicted score $pred(u_x, item_i)$ using the CF algorithm, as shown in equation (4).

In this work we consider the task of recommending top k items, *i.e.*, k items having the highest predicted scores, which maximize $\prod_{i=1}^{k} pred(u_x, p_i)$. Note that the *general* strategy generates one list of recommendations for all users, the group-based *aggregated models* and *aggregated predictions* strategies generate one list for each family, and the *personalized* strategy generates one list for each user.

3.2 Weighting Models

When aggregating the data of individual users, it is natural to allow for some users to have more influence than others. In this way, users who are seen to have authority or who are trusted, are treated differently in order to impact the recommendation process. Authority and influence can be determined either through explicit ratings or through implicit contribution or consumption measures. Hence, aggregated group-based data can be achieved by weighting the data of individual users accordingly.

We investigate four models for weighting user data. The first two are static and assign to users pre-defined weights. The *uniform* model weights users uniformly, *i.e.*, $\omega(u_x, f_a) = 1$. The *heuristic* model is role-based, where a *role* refers to a user's function within a family: *applicant*, *partner*, or *child*. The model presumes that $\omega(u_x, f_a)$ is defined solely by the user's role. An applicant's weight is $\omega(u_x, f_a) = 0.5$, as they are likely to be highly engaged with the content, a partner's weight is $\omega(u_x, f_a) = 0.3$, as they are likely to be reasonably engaged, and a child's weight is $\omega(u_x, f_a) = 0.1$, as they are not likely to be engaged.

Two other weighting models are based on the observed user interactions with the content. The weights assigned to users reflect their activity $act(u_x)$, *i.e.*, number of ratings $rat(u_x, item_i)$, as a predictor of their degree of engagement. The *role-based* model weights users according to the activity $act(u_x)$ of users in the same role across the entire community, as shown in equation (6). The *family-log* model weights users according to their activity $act(u_x)$ in relation to other family members $u_y \in f_a$, as shown in equation (7).

$$\omega(u_x, f_a) = \frac{\sum_{y \in U} act(u_y) \mid role(u_y) = role(u_x)}{\sum_{y \in U} act(u_y)} \quad (6)$$

$$\omega(u_x, f_a) = \frac{act(u_x)}{\sum_{y \in f_a} act(u_y)} \quad (7)$$

3.3 Switching Hybridization

It has been shown that in individual CF recommendations, no single recommendation strategy is generally superior to all others. On the contrary, the best performance is achieved when several strategies are *hybridized* in order to better match the recommendation request [5]. The hybridization can be achieved in many ways, *e.g.*, by merging the predicted scores, user features, or recommendation algorithms. We posit that this observation is true also in the group-based recommendations and hybridize the strategies presented in Section 3.1.

Initial evaluation of group recommendations showed that the *personalized* strategy achieved highest accuracy but lowest coverage, the *general* strategy achieved lowest accuracy but highest coverage, while the performance of the group-based strategies was moderate [2]. To hybridize these strategies, we developed a *switching* hybridization strategy, which "*switches between recommendation techniques depending on the situation*" [5]. The criterion for a strategy selection is the density of the users' data[1]. We quantify the density degree $dens(u_x)$ as the ratio between the number of items that were rated by u_x and overall number of items. Hence, the switching of strategies is defined as shown in equation (8).

$$strat(u_x) = \begin{cases} general & 0 \leq dens(u_x) < \beta_1 \\ group-based & \beta_1 \leq dens(u_x) < \beta_2 \\ personalized & \beta_2 \leq dens(u_x) < 1 \end{cases} \quad (8)$$

β_1 and β_2 denote the density thresholds for switching between the *general* and the group-based, and between the group-based and *personalized* strategies, respectively.

3.4 Extreme Case Heuristics

In case of extremely positive or negative data, the models presented in Section 3.2 may be inapplicable. For example, consider an extremely negative recipe rating provided by a family member. Even if the ratings of other members are positive, the recipe should not be recommended as the user is not likely to eat the recommended meal. Alternatively, if a family member provided extremely positive rating for a recipe, other family members may also like this meal[2].

To prevent such situations, we enhance the weighting models by introducing two data aggregation heuristics [11]. When aggregating the individual data into group-based data, the *least misery* heuristics assigns $\widetilde{\omega}(u_x, f_a) = 1$ to the user, who provided the extremely negative data, and $\widetilde{\omega}(u_y, f_a) = 0$ to other family members $u_y \in f_a$. Otherwise, a normal weighting model is applied. Note that the *least misery* heuristics is applicable when aggregating both individual ratings $rat(u_x, item_i)$ into family-based rating $rat(f_a, item_i)$ (equation 2) and individual predictions $pred(u_x, item_i)$ into family-based prediction $pred(f_a, item_i)$ (equation 5). The *least misery* heuristics is defined as shown in equation (9). γ_1 denotes the threshold for considering a rating as extremely negative.

$$\widetilde{\omega}(u_x, f_a) = \begin{cases} 1 & rat(u_x, item_i) \leq \gamma_1 \\ \omega(u_x, f_a) & \text{otherwise} \end{cases} \quad (9)$$

Similarly, we define the *most pleasure* heuristic. When aggregating the individual data into group-based data, the *most pleasure* heuristic assigns $\widetilde{\omega}(u_x, f_a) = 1$ to the user, who provided the extremely positive data, and $\widetilde{\omega}(u_y, f_a) = 0$ to other family members $u_y \in f_a$. The *most pleasure* heuristics is defined as shown in equation (10). γ_2 denotes the threshold for considering a rating as extremely positive.

$$\widetilde{\omega}(u_x, f_a) = \begin{cases} 1 & rat(u_x, item_i) \geq \gamma_2 \\ \omega(u_x, f_a) & \text{otherwise} \end{cases} \quad (10)$$

4. EVALUATION

An evaluation was carried out using a dataset of explicit ratings for recipes, gathered during a study observing interaction of families with an experimental eHealth portal.

[1] See [5] for other switching criteria.

[2] Cases, in which one of the ratings is extremely positive and another extremely negative are out of scope of this work.

Table 1: Experimental Dataset

N_{users}	N_{fam}	$N_{fam,n=1}$	$N_{fam,n=2}$	$N_{fam,n=3}$	$N_{fam,n=4}$	N_{items}	$N_{rat(u,i)}$	$density$
170	108	70	18	12	7	136	3305	14.38%

Table 2: Comparison of Recommendation Strategies

	$general$	$agg-mod$	$agg-pred$	$personalized$	$significance$
F1	0.1687	0.2266	0.1821	0.3368	**p<0.05**
MAE	0.2163	0.1860	0.2102	0.1746	**p<0.01**
coverage	100%	97.63%	93.75%	85.41%	p>0.05

Figure 2: Recipe rating interface

The aim of the analysis was to uncover a recommendation strategy, which would be most appropriate to implement in a group-based recommender. Specifically, we aimed to compare the accuracy of two group-based recommendation strategies, four weighting models, and assess the impact of switching hybridization, extreme case heuristics, and group characteristics on the performance of group recommendations. Partial results obtained for a substantially smaller dataset of implicit browsing logs were presented in [2].

4.1 Experimental Setting and Metrics

The dataset was gathered over a three week period in July 2009. During this period a dataset of explicit symbolic ratings on a 5-Likert scale ranging from *hate* to *love* was captured from participants for a corpus of recipes sourced from the CSIRO Total Wellbeing Diet book [14]. The symbolic ratings were converted into numeric ratings ranging from 1 to 5. Figure 2 depicts the rating interface.

Table 1 summarizes the dataset. The columns represent the overall number of users, overall number of families, number of families in which n=1, 2, 3, or 4 users provided ratings[3], number of recipes in the dataset, number of ratings captured, and density of the data (ratio between the number of captured and possible ratings). The distribution of recipe ratings was not uniform: 883 were rated *hate*, 1352 - *don't like*, 741 - *neutral*, 254 - *like*, and 75 - *love*.

For each user/family, a one-off similarity matrix with other users/families was computed using Cosine Similarity [10]. Using these matrices, *N=5* most similar users/families were selected, leave-one-out recipe rating predictions were computed, and recommendations were generated. The recommendations were evaluated against the ratings provided by individual users using the *F1*, *precision@k*, Mean Absolute Error (*MAE*), and *coverage* metrics [8].

Let us denote by $\mathbb{V}$ the set of positive recipes rated *neutral*, *like*, or *love*, and by $\mathbb{R}$ the set of recipes with positive predicted scores of 3 or higher. Hence, *precision* of the recommendations is computed by $\frac{|\mathbb{V}\cap\mathbb{R}|}{|\mathbb{R}|}$ and *recall* by $\frac{|\mathbb{V}\cap\mathbb{R}|}{|\mathbb{V}|}$. When the size of $\mathbb{R}$ is k, the precision metric is referred to as *precision@k*. Combining the precision and recall metrics yields the *F1* metric, which represents their harmonic mean with equal weights, as shown in equation 11.

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{11}$$

MAE is computed as the average difference between the predicted and provided rating for user u_x and item $item_i$, normalized by the cardinality of the range of ratings $R_d \in [1..5]$ in the dataset, as shown in equation 12.

$$MAE = \sum_{x\in U}\sum_{i\in I} \frac{\mid pred(u_x, item_i) - rat(u_x, item_i) \mid}{\mid R_d \mid} \tag{12}$$

Another metric was the *coverage* of the recommendations. It reflects the relative portion of items, for which an algorithm successfully generated recommendations (regardless of their accuracy). It is computed by dividing the number of items for which a prediction was generated by the overall number of items in the dataset.

4.2 Recommendation Strategies

The first question relates to comparative performance of the recommendation strategies presented in Section 3.1 and, in particular, of the two group-based strategies. Table 2 presents the average predictive accuracy MAE score, classification accuracy F1 score, and coverage obtained for the four recommendation strategies: *general*, *aggregated models*, *aggregated predictions*, and *personalized*. The right-most column focuses on the group-based *aggregated models* and *aggregated predictions* strategies and presents whether the difference between the two is statistically significant[4]. Uniform weighting is applied, *i.e.*, $\omega(u_x, f_a) = 1$.

The results show that the *aggregated models* strategy outperformed the *aggregated predictions* strategy across all metrics: higher F1, lower MAE, and higher coverage scores. The difference was significant for F1 and MAE, respectively, p<0.05 and p<0.01, but not significant for coverage.

A single F1 score hides too much information about the classification accuracy. We measured precision-recall scores obtained by each user for each recommendation strategy. Figure 3 depicts overall polynomial regression curves of the strategies generated form the individual user scores[5]. The graph shows that the *aggregated models* strategy outperformed the *aggregated predictions* strategy, as the former has a greater area under the curve than the latter. The difference was significant, p<0.01.

[3]Families having 1 active user were excluded from the testing set and used only in the training set.

[4]All statistical significance results hereafter refer to a two-tailed t-test assuming equal variances.

[5]For the sake of clarity, we omit individual precision-recall scores and plot only the regression curves.

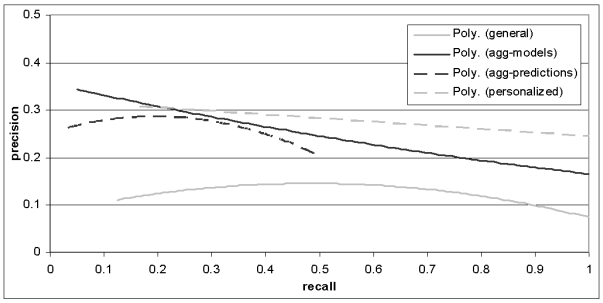

Figure 3: Precision-recall of strategies

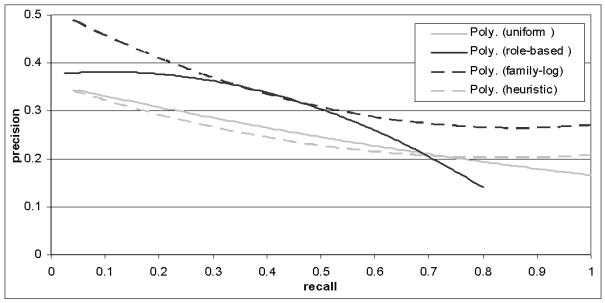

Figure 5: Precision-recall of weighting models

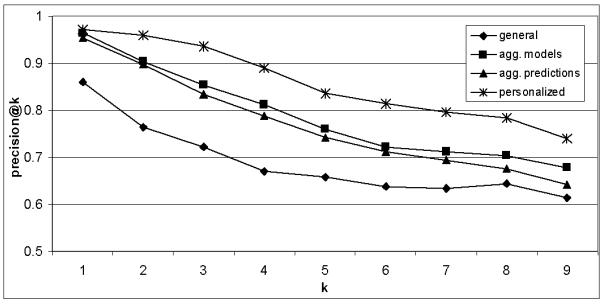

Figure 4: Precision@k of strategies

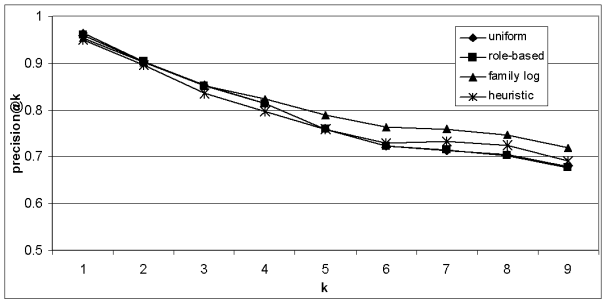

Figure 6: Precision@k of weighting models

Figure 4 depicts the average precision@k scores obtained by the recommendation strategies for $k \in [1..9]$. The graph shows that for any value of k the *aggregated models* strategy obtained higher precision@k score than the *aggregated predictions* strategy. The difference was not significant.

In summary, all the comparisons between the group-based recommendation strategies show that the *aggregated models* strategy is superior to the *aggregated predictions* strategy. This is due to the reliable models that the *aggregated models* strategy creates, which facilitate generation of accurate recommendations. Practically, this means that group-based recommendations should be generated by aggregating individual models into group models and then using these models in the recommendation process.

4.3 Weighting Models

The second question relates to comparative performance of the weighting models presented in Section 3.2 and, in particular, of the two interaction-based models. Table 3 presents the average predictive accuracy MAE score, classification accuracy F1 score, and coverage obtained for the four weighting models: *uniform*, *role-based*, *family-log*, and *heuristic*. The right-most column focuses on the interaction-based *role-based* and *family-log* models and presents whether the difference between the two is statistically significant. The evaluation used the *aggregated models* recommendation strategy, which was discovered to be the most appropriate.

The performance of the models can be partitioned into two groups: static *uniform* and *heuristic* models and interaction-based *role-based* and *family-log* models. The results show that the interaction-based models outperformed the static models across both accuracy metrics: higher F1 and lower MAE scores. The impact of weighting models on the coverage was negligible. Comparison of the two interaction-based models shows that the *family-log* model outperformed the *role-based model* model across both accuracy metrics: higher F1 and lower MAE scores. The difference was significant for MAE, $p<0.05$, but not significant for F1.

Similar to previous experiment, Figures 5 and 6 depict, respectively, the overall polynomial regressions curves of the discrete precision-recall scores and average precision@k scores obtained by the weighting models. Figure 5 clearly differentiates between precision-recall curves of static and interaction-based models. The interaction-based *role-based* and *family-log* models outperformed the static *uniform* and *heuristic* models. Comparison of the interaction-based models shows that *family-log* model outperformed the *role-based model* model. The difference was significant, $p<0.05$.

The difference between the precision@k curves of the models in Figure 6 is less pronounced. For low k, the weighting models were comparable: for $k = 1$ all four models obtained a similar precision score. The models separated at $k = 3$, with the static models becoming less accurate. Eventually, interaction-based models outperformed the static models, with the *family-log* model obtaining the highest precision@k score. The difference between the *family-log* and *role-based* models was not significant.

In summary, weighting models assigning weights according to the observed user interactions are superior to models assigning static weights. Between the two interaction-based models, the *family-log* model is superior to the *role-based* model. This is due to the localized nature of the *family-log* model, such that the weights reflect interactions observed within the family only, rather than within the entire community. Practically, this means that the weights assigned to users' data should reflect family-based or community-based interactions (in this order of priority), while predefined weighting models should be avoided.

Table 3: Comparison of Weighting Models

	$uniform$	$role-based$	$family-log$	$heuristic$	$significance$
F1	0.2266	0.2583	0.2662	0.2225	p>0.05
MAE	0.1860	0.1841	0.1830	0.1851	**p<0.05**
coverage	97.63%	97.63%	97.62%	97.65%	p>0.05

Table 4: Impact of Switching Hybridization

	$agg-mod$	$personalized$	$switching$	$significance_{agg-mod}$	$significance_{pers}$
F1	0.2662	0.3368	0.3370	**p<0.01**	**p<0.05**
MAE	0.1830	0.1746	0.1714	**p<0.05**	**p<0.05**
coverage	97.62%	85.41%	99.03%	p>0.05	**p<0.01**

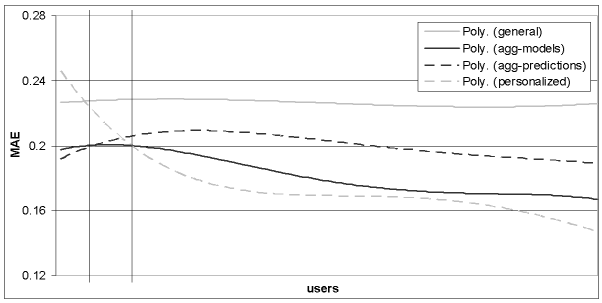

Figure 7: MAE of strategies

4.4 Switching Hybridization

The third question relates to the impact of the switching hybridization strategy presented in Section 3.3. In order to determine the switching thresholds β_1 and β_2, we used the *family-log* weighting model, which was discovered to be the most appropriate, and compute for each user the MAE score for the four recommendation strategies: *general*, *aggregated models*, *aggregated predictions*, and *personalized*.

Figure 7 depicts overall polynomial regression curves of the strategies generated form the individual MAE scores. The users are arranged in increasing order of data density, *i.e.*, left-most users provided the lowest number of ratings and right-most – the highest . Behavior of the strategies is inline with previous CF research [10]. The MAE of the *general* strategy remains roughly unchanged, while that of the *aggregated models*, *aggregated predictions*, and *personalized* strategies decrease as the data density increases. Our aim, to uncover the most appropriate switching thresholds, should lead to a reduction in the overall MAE.

Initially, the *aggregated predictions* strategy demonstrates the lowest MAE outperforming the *aggregated models* strategy. We posit that this happens because the aggregation of individual predictions at this level of density is more accurate than the aggregated model of a family. At $\beta_1 = 5$ ratings (corresponds to 12% density[6]), the aggregated family models are sufficiently accurate and the *aggregated models* outperforms the *aggregated predictions* strategy. Finally, at $\beta_2 = 8$ ratings (corresponds to 19% density), the individual user models are accurate enough and the *personalized* strategy outperforms the group-based strategies. The thresholds are marked in Figure 7 with vertical lines.

[6]In a practical recommender system a user is unlikely to rate all the items. Hence, relative density of 100% refers the highest number of provided ratings (in our case – 42).

We applied the derived β_1 and β_2 thresholds as the switching criteria and evaluated the performance of the switching strategy. Table 4 presents the average MAE, F1, and coverage scores obtained by the *aggregated models* strategy with the *family-log* weighting model (the most appropriate group strategy), the *personalized*, and switching hybridization strategy. The two right-most columns focus on the switching strategy and present whether the difference between it and, respectively, the *aggregated models* and *personalized* strategies, is statistically significant.

The results show that the switching strategy clearly outperformed *personalized* strategy. The differences in the obtained F1 and MAE scores were significant, p<0.05. Note the improvement obtained for coverage. It increased from 85.41% to 99.03%, showing that switching successfully applied the group-based strategies, when the data were insufficient for personalized recommendations. The difference was significant, p<0.01. As expected, the switching strategy also outperformed the *aggregated models* strategy. The difference in F1 and MAE was significant, respectively, p<0.01 and p<0.05. The difference in coverage was not significant.

In summary, the best performing switching strategy applied the *aggregated predictions* strategy for the lowest density of users data, then the *aggregated models* strategy, and the *personalized* strategy in the majority of cases. The switching strategy was discovered to be superior to all the individual strategies across both the accuracy metrics and obtained extremely high coverage of recommendations.

4.5 Extreme Case Heuristics

The fourth question relates to the impact of the *least misery* and *most pleasure* heuristics presented in Section 3.4. Since the distribution of recipe ratings was not uniform, we considered the 254 *like* and the 75 *love* (in total, 329) ratings as extremely positives and randomly selected 329 (out of the 883) *hate* ratings as extremely negatives[7]. Hence $\gamma_1 = 1$ and $\gamma_2 = 4$. Table 5 presents the average MAE, F1, and coverage scores obtained by the *aggregated models* strategy with the *family-log* weighting model (the most appropriate group strategy), and the same strategy using, respectively, the *least misery* and *most pleasure* heuristics. The two right-most columns focus on the heuristics and present whether the difference introduced by the heuristics is statistically significant.

Overall, the *least misery* and *most pleasure* heuristics decrease the accuracy of recommendations. The F1 score decreases (not significant for *least misery*, significance for *most*

[7]Personalized extremeness thresholds will be investigated in the future.

Table 5: Impact of Extreme Case Heuristics

	$agg-mod$	$agg-mod_{LM}$	$agg-mod_{MP}$	$significance_{LM}$	$significance_{MP}$
F1	0.2662	0.2549	0.2457	p>0.05	**p<0.05**
MAE	0.1830	0.1847	0.1853	**p<0.05**	**p<0.05**
coverage	97.62%	97.61%	97.61%	p>0.05	p>0.05

pleasure, p<0.05) and MAE increases (significance for both, p<0.05). The coverage also decreases, but not significant for both the heuristics.

Breaking down the F1 score into precision and recall, we observed that the *least misery* heuristic affected the recall of recommendations. This is due to the fact that a user's positive rating for a recipe could be outweighed by an extremely negative rating of another family member, which prevents this recipe from being recommended and decreases the recall. On the contrary, the *most pleasure* heuristics affected the precision, as some negatively rated recipes could be recommended to a user due to an extremely positive rating from another family member. The decrease in recall for the *least misery* heuristic and of precision for the *most pleasure* heuristics were significant, p<0.05.

The accuracy decrease, however could be balanced by a higher user appreciation of recommendations, which is not measurable in an offline evaluation. For example, in the *least misery* heuristics, recipes hated by a group member are unlikely to be recommended to others, which may increase system trust. Similarly, in the *most pleasure* heuristics, recipes loved by a group member are likely to be recommended to others, which may increase serendipity.

In summary, the extreme case heuristics negatively affected the accuracy of recommendations. However, they have the potential to positively affect user appreciation and serendipity of the recommendations provided by the system, which could not be measured in our evaluation. We posit that the heuristics can be applied when the degree of confidence in the recommendation list is high, such that removing certain items will not severely damage it, but including certain items can potentially sustain user engagement.

4.6 Group Characteristics

The fifth question relates to the differences in the performance of the recommendation strategies across various groups. In particular, we evaluate the impact of two group characteristics on the accuracy of generated recommendations: size and homogeneity of a group. In both cases, we used the *family model* strategy and *family logs* weighting model, as it was discovered to be the most appropriate group strategy.

Firstly, we analyze the dependency between the size of a family and accuracy of recommendations. Figure 8 shows the average MAE scores obtained for various families. The families are arranged in a decreasing order of MAE, and are color-coded according to the number of members: white bars represent 2 user families, grey – 3 user families, and black – 4 user families. The accuracy of recommendations mainly increases with the family size. Most families with high MAE are 2 user families, while more 3 and 4 user families, *i.e.*, more grey and black bars, occur as MAE increases. The correlation between the MAE score and the number of family members is −0.644. Hence, the accuracy of recommendations improves with the number of family members

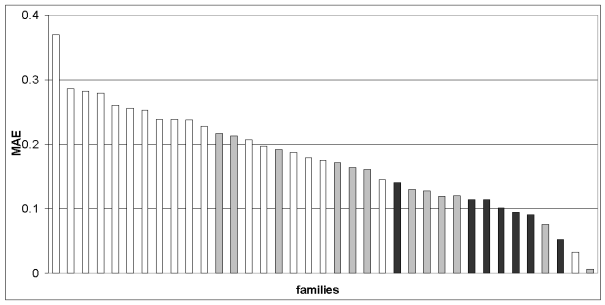

Figure 8: MAE vs. family size

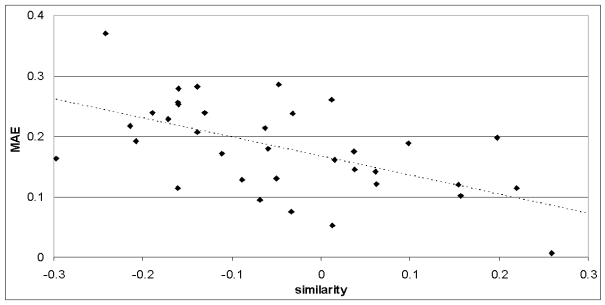

Figure 9: MAE vs. intra-family similarity

and amount of data available, as the data of large families are denser than of small families.

Secondly, we analyze the dependency between the similarity of family members and accuracy of recommendations. Figure 9 shows the MAE scores obtained for various families as a function of the average similarity of family members. The accuracy of recommendations increases with the similarity of members: MAE of families having low similarity of members is high and it decreases as the similarity increases. The correlation between the MAE score and the average similarity is −0.628. Hence, the accuracy of recommendations improves with similarity of members and homogeneity of families, as the data of homogeneous families are more reliable than of non-homogeneous families.

In summary, the performance of group-based recommendations was discovered to depend on group characteristics. Particularly, the accuracy of recommendations increases with the size an homogeneity of groups.

5. CONCLUSIONS

With the dissemination of recommender technologies, more and more scenarios evolve, in which group-based recommendations, addressing a group of users rather individuals, need to be provided. This work focuses on family-based CF recipe recommendations, a particular case of group recommendations. The grouping of family members in this case is inher-

ent, while recipes offer a natural case for group recommendations, as family members consume joint meals. Hence, a family-based recipe recommender should consider and satisfy preferences of all the members and not recommend a recipe that is likely to be rejected by some.

In this work we focused on uncovering the most appropriate group recommendation strategy and this was achieved in several steps. First, we focused on the most appropriate recommendation strategy and user weighting model. Our evaluation showed that the best performance of group recommendations is obtained when individual user models are aggregated into group-based models, which are then used in the recommendation process. The individual data of group members need to be aggregated in a weighted manner, such that the weights reflect the observed interaction of group members, focusing on interactions observed with as localized as possible boundaries.

Also, we evaluated a switching hybridization strategy, which selects a recommendation strategy to apply according to the user data density. The results showed that the accuracy of the switching strategy is superior to all individual recommendation strategies, and it demonstrated very high coverage score. Next, we evaluated the performance of two extreme case heuristics. These were discovered to decrease the accuracy of recommendations, but could potentially improve user appreciation and sustain user engagement. Finally, we discovered that the performance of group-based recommendations depends on group characteristics, in particular, on the size and homogeneity of the groups.

Hence, when generating group-based CF recommendations, the system should initially determine the recommendation strategy to apply. If the available user data are sufficient, personalized CF recommendation should be applied. Otherwise, a group-based strategy should be applied as follows: (1) individual user models of the group members should be aggregated into group-based models, (2) weights assigned to individual user models should reflect the observed importance of users, (3) the aggregated models should be used in a family-based CF recommendation process, and (4) group-based recommendations should be delivered to the group members. When aggregating individual models, extreme case heuristics can be applied. Although the latter can slightly decrease the accuracy of recommendations, they can increase user appreciation and system trust.

In the future, we plan to investigate sequential group-based recommendations. Often, recommendations are not provided on an ad-hoc basis, but users have prolonged interactions with the system. It is important to handle such interactions differently, *e.g.*, compensate users, whose satisfaction was low in past interactions. Also, we plan to investigate group-based dynamics. Different groups may have complex social intra-group relationship. We will investigate how these relationships, *e.g.*, roles, dominance, and decision taking, affect group recommendations. Finally, we plan to conduct a similar evaluation in other domains to verify that the outcomes of this work are generalizable.

6. ACKNOWLEDGMENTS

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

7. REFERENCES

[1] L. Ardissono, A. Goy, G. Petrone, M. Segnan, and P. Torasso. Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices. *Applied Artificial Intelligence*, 17(8-9):687–714, 2003.

[2] S. Berkovsky, J. Freyne, and M. Coombe. Aggregation trade offs in family based recommendations. In *Australasian Conf. on Artificial Intelligence*, 2009.

[3] S. Berkovsky, T. Kuflik, and F. Ricci. Distributed collaborative filtering with domain specialization. In *Conf. on Recommender Systems*, 2007.

[4] P. Brusilovsky, G. Chavan, and R. Farzan. Social adaptive navigation support for open corpus electronic textbooks. In *Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems*, 2004.

[5] R. D. Burke. Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12(4):331–370, 2002.

[6] Y.-L. Chen, L.-C. Cheng, and C.-N. Chuang. A group recommendation system with consideration of interactions among group members. *Expert Systems with Applications*, 34(3):2082 – 2090, 2008.

[7] J. Freyne and B. Smyth. Cooperating search communities. In *Int. Conf. on Adaptive Hypermedia and Adaptive Web-Based Systems*, 2006.

[8] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl. Evaluating collaborative filtering recommender systems. *Transactions on Information Systems*, 22(1):5–53, 2004.

[9] A. Jameson and B. Smyth. Recommendation to groups. In P. Brusilovsky, A. Kobsa, and W. Nejdl, editors, *The Adaptive Web: Methods and Strategies of Web Personalization*. Springer, 2007.

[10] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon, and J. Riedl. GroupLens: applying collaborative filtering to Usenet news. *Communications of the ACM*, 40(3):77–87, 1997.

[11] J. Masthoff. Group modeling: Selecting a sequence of television items to suit a group of viewers. *User Modeling and User-Adapted Interaction*, 14(1):37–85, 2004.

[12] J. F. McCarthy and T. D. Anagnost. Musicfx: An arbiter of group preferences for computer aupported collaborative workouts. In *Int. Conf. on Computer Supported Collaborative Work*, 1998.

[13] K. McCarthy, L. McGinty, and B. Smyth. Case-based group recommendation: Compromising for success. In *Int. Conf. on Case Based Reasoning*, 2007.

[14] M. Noakes and P. Clifton. *The CSIRO Total Wellbeing Diet Book*. Penguin, 2005.

[15] M. O'Connor, D. Cosley, J. A. Konstan, and J. Riedl. Polylens: a recommender system for groups of users. In *European Conf. on Computer Supported Cooperative Work*, 2001.

[16] Z. Yu, X. Zhou, Y. Hao, and J. Gu. Tv program recommendation for multiple viewers based on user profile merging. *User Modeling and User-Adapted Interaction*, 16(1):63–82, 2006.